Exploring Neural Correlates of Flexible Cognition During a Complex Decision Making Task

John Ferré

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy

University of Washington
2023

Reading Committee:
Adrienne L. Fairhall, Chair
Elizabeth A. Buffalo
J. Nathan Kutz

Program Authorized to Offer Degree:
Department of Physics

University of Washington

Abstract

Exploring Neural Correlates of Flexible Cognition During a Complex Decision Making Task

John Ferré

Chair of the Supervisory Committee:
Adrienne L. Fairhall
Department of Physiology and Biophysics

Efficiently solving complex decision making tasks typically arises from the use of schemas, abstract structures that organize experience and guide integration of novel information. There is a wealth of knowledge related to how the hippocampus, the prefrontal cortex, and communication between the two areas relate to schema formation, decision making, memory encoding, reward representations and behavioral planning. Less focus, however, has been placed on tasks involving ambiguous credit assignment, where feedback cannot be uniquely attributed to a single feature. In this case, information associating rewards with multiple stimulus features needs to be integrated across choice history to identify which features give rise to specific rewards. How does neural circuitry support a learned schema in complex credit assignment problems?

To understand the neural dynamics, I focus on data from non-human primates solving a multivariate bandit task involving ambiguous feedback of stimulus features, recorded in the Buffalo lab at UW. Through behavioral modeling, I find that non-human primates solve the problem with a reinforcement learning framework, suggesting schema-like learning by biasing choices towards recently rewarded features. Behavioral modeling allows us to fit an estimate of the belief state of the non-human primate, which we leverage to seek encoding of abstract feature variables in the hippocampus, prefrontal cortex, and basal ganglia. Particularly, decoding of abstract feature variables during the feedback period, along with multiple timescales of feedback specific modes, suggest that the neural circuitry updates belief states

by integrating feedback information with working memory. Further, bidirectional information flow between the hippocampus and prefrontal cortex provides evidence for the hypothesis that the hippocampus guides neocortical plasticity while the prefrontal cortex stores and retrieves memories. Altogether, these findings suggest possible dynamical representations for computations following a reinforcement learning framework, and these findings lay the groundwork for understanding the underlying circuitry that supports schema representations.

Acknowledgements

I would like to thank everyone who has helped and supported me during my academic journey, both professionally and personally.

I would like to thank my advisor, Adrienne Fairhall, for their advice and support. I also would like to thank Ariel Rokem for help during the early and middle stages of my research: they helped in numerous ways, from learning the basics of neuroscience to help with developing the NS-DMD method. I'd also like to thank Edgar Walker, who greatly helped during the latter half of my research. Thanks to Elizabeth Buffalo for the advice, support, and use of data in this thesis. Thanks to Nathan Kutz for support, feedback, and suggestions in a variety of manners and in regards to the NS-DMD method.

I'd especially like to thank those heavily involved in the weeds of the U19 project. Thanks to Mike Jutras, who has helped with a plethora of analysis and pre-processing of the data. Thanks to Patrick Zhang, Po-Chen Kuo, Ellen Streitwieser, and Michael Mendelson for in depth discussions and analysis. Thanks to Vishwa Goudar and Jeong Woo Kim (Brian) for feedback and suggestions from a theoretical background. Thanks to Ivan Skelin, Barbara Peysakhovich, Vinay Shirhatti, Haoxin Zhang, Ruby Marx, Eduardo Sandoval, Jie Zheng, and Bhargavi Ram for their feedback, suggestions, and ideas.

I'd like to pay special thanks to those involved with the data collection and processing of the data. Thanks to the Buffalo and Knight labs for collection of behavioral data. Thanks to Adam Dede, Sufia Ahmad, Sierra Schleufer, and Ian O'Leary for collection of the primary neural data used in the analyses in this thesis. Thanks to Mike Jutras spike sorting all of the neural data to allow for single unit analysis.

I'd like to thank Jon Rueckemann, Charlie Gray, Steven Hoffman, and David Kleinfeld for their advice/suggestions early on in my research. Thanks to Bing Brunton for advice concerning the NS-DMD paper. I would also like to thank the numerous people who have given a variety of feedback and suggestions during the U19 meetings: Alison Mehravari, Anna Jafarpour, Ben Grannan, Robert Knight, Ian O'Leary, David Freedman, Jack Lin, Kurt Weaver and Peter Brunner. I'd further like to thank those in the Computational Neuroscience Center, who have given a plethora of feedback on all analyses: Alison Duffy, Leenoy Meshulam, Belle Liu, Daniel Zdeblick, David Bell, Divyansh Gupta, Doris Voina, Monica Liu, Fereshteh Lagzi, Jessica Huszar, Eric Shea-Brown, Faezeh Aminmansoor, Helena Liu, Hengji Wang, Ilse Dippenaar, Praveen Venkatesh, Scott Sterrett, Stefano Recanatesi, and Ziyu Lu. I'd also like to thank Daniel Sitonic and Erik Sundell for their technical support. And of course, I would like to thank Samu Taulu, Marcel den Nijs, and Ali Shojaie for their help and suggestions.

Thanks to the funding agencies that made this all possible. This work was supported by the NIH BRAIN Initiative U19NS107609 and the Simons Collaboration for the Global Brain.

Thanks to my girlfriend Alisha Husain, our cat Baela, my parents Julia Ferré and Carl Ferré, my brothers Franz Ferré, Nels Ferré, and Gus Ferré, and sister-in-laws Jamie Ferré and Sakura Yoshino for all of their love and support. Thanks to my close friends Roland Farrell, Zack Aemmer, Nick Du,

Charlotte Zimmerman, Brandon Montemuro, John Goldak, Patrick Zhang, Leif Hasel, Taraneh Mahjour, Jack Barlow, and Charles Lewis for their support, especially during the difficult covid years. I'd also like to thank Leenoy Meshulam and Monica Liu for advice and support towards the end of my Ph.D.

Contents

Chapter 1

Introduction

Decision making is a critical paradigm for success in the natural world. Individuals learn from the success or failure of previous actions to improve their chances of future rewards. Difficulty increases when the number of features and ambiguity increases. For example, many features of berries, such as color, texture, or environmental conditions could correlate with its ripeness. Any animal interested in identifying ripe berries needs to navigate the feedback received from all previously consumed berries to uncover the underlying features that best correlate with ripeness. Not only is this problem difficult due to the sheer number of features and samples, but also individuals encounter the credit assignment problem [1], where they need to assign credit to only the subset of features that they believe is correlated to the ripeness.

One framework for understanding the beliefs of individuals is through the concept of schemas [2], high level structure that helps to organize information. Schemas allow one to place new information within a context, which leads to efficient processing of difficult tasks. For example, the hippocampus contains cells that encode space, known as place cells; this can create a schema-like representation [3], where place cells are typically used to encode spatial information [1, 4, 5]. Schemas are expected to be useful in high dimensional credit assignment problems, where generalization across features can be used to efficiently organize incoming information.

While schemas are thought to form in the hippocampus [6], growing evidence has shown an interaction between the hippocampus and prefrontal cortex in difficult tasks [2]. One hypothesis is that the hippocampus is responsible for forming "declarative" memories [7], or memories that require specific actions by the brain. Motor skills or habits are not considered declarative. In this sense, the hippocampus may be responsible for how information is encoded, while the prefrontal cortex may be responsible for holding and manipulating that information [8]. The communication between the two areas is critical in difficult decision making tasks, where interactions are necessary for efficient memory encoding.

With the importance of the hippocampus and PFC in mind, the overarching goal of this thesis is to

understand the neural representations necessary for individuals that are solving the naturalistic problem of high dimensional credit assignment. With experimental data collected from the Buffalo lab, there are numerous difficulties in uncovering neural representations. While it's believed that a schema guides the decision-making process, the schema itself is not known to the experimenter. The first aim is to behaviorally model subjects' choices such that there is an understanding of the subjects' behavior and their internal beliefs. The second aim is to find neural evidence for these behavioral models, which would provide evidence toward the use of a schema while solving credit assignment problems. Finally, the third aim is to uncover the representations necessary in solving these tasks. With the behavioral models, one can uncover the neurons and brain areas responsible for computations and schema representations. The brain-wide computations and communication then help explain the individuals' choices.

1.1 Schemas

The terms "schema," "learning set," "cognitive map," and "mental model" have somewhat similar meanings. Cognitive maps [9] are typically used in spatial tasks to describe how a subject maps relative locations of objects in an environment [10]. They have also been used in non-spatial organizations [11], such as the cognitive mapping of frequencies [12]. Schemas are more general than cognitive maps, as they are frameworks that provide default beliefs or assumptions when new information is provided [13]. A spatial schema can be thought of as a framework to view and store information about similar environments without focusing on any specific location [10]. As an example, a cognitive map is created when moving to a new city to understand the relative locations of important areas. A schema is used to quickly interpret, identify, and categorize locations; e.g. cities typically include a downtown area, neighborhoods, sometimes a central highway to transport one self large distances quickly, and/or rivers/waterways that naturally divide areas. However, schemas apply to more general situations than just spatial maps, such as in memory formation and decision making situations [2, 14]. For example, schemas could be used when forming associations of main and side characters in movies or tv shows [15]. On a side note, this may be one reason why experts in one area (e.g. science) can quickly learn adjacent areas (e.g. math). Generally speaking, schemas are useful for taking advantage of common structures in the environment, which can be useful for parsing novel information.

Mental models, sometimes referred to as internal models, are similar in concept to schemas. Mental models are specific models of the world that the individuals create to guide processes [16], and they can be updated upon performing actions and receiving feedback. This underpins the concept of predicative coding, where individuals anticipate future feedback based on prior knowledge [17]. These models are explicit views of the state of the world as opposed to the implicit assumptions of the world formed by schemas. A cognitive map, for example, could be thought of as a mental model, and a schema could

give rise to the formation of specific mental models. One natural case of mental models occurs in motor control and trajectory planning [18], where an internal model guides body parts to the desired location. Quantitative cognitive models, models of subjects behavior that can be used to correlate to neural data [19], can be used as hypotheses for individuals mental models [20]. This approach leads to interpretations of the individuals choices along with a hypothesis of the underlying mental model.

Lastly, schemas share similarities with learning sets [21], where subjects "learn to learn" associations very quickly after multiple problems. E.g. monkeys, when given multiple pairwise discrimination problems, eventually learn to one-shot discriminate pairs of objects and rewards [21]. In developing both schemas and learning sets, subjects improve their ability to learn new problems. However, learning sets are less generalizable to new situations.

How Have Schemas Been Applied?

There has been growing research into the form of schemas when rapidly forming connections in contexts of decision making, paired associations, and spatial organization [3, 22, 23]. In one example, Tse et al. [22] found that rats form paired associations more quickly after many learned paired associations, a memory consolidation process made possible by developing a schema based on statistical regularities. In their case, rats formed a spatial memory of the arena, which could be then used to rapidly assign locations to novel stimuli. They found that the hippocampus was important in this process, as hippocampal-lesioned rats failed to form these associations. However, expert rats, trained for an extraordinary amount of time, did not perform worse when the hippocampus was lesioned. This indicates that the hippocampus was important for the development of the schema.

In a similar example, Morris et al. [24] found that rats learn a spatial map of different food locations in an arena across the course of a day. After training, rats are able to one-shot learn new locations of food. They argue that a schema is learned such that the rats can efficiently integrate new knowledge and prevent rapid forgetting. The schema also likely plays a role in memory consolidation. They argue that a learning set hypothesis is less likely, since expert animals learn the new set of food locations much slower when a different arena is introduced.

Zhou et al. [23] hypothesize that learning tasks with schema lead to similar neural representations across different problems. They find that after rats learn multiple odor sequence problems, the neural representations in the orbital frontal cortex are similar across problems with the same underlying structure. They believed that generalization occurred across problems with the use of a schema.

In relation to cognitive maps, Baraduc et al. [3] found cells in the monkey hippocampus that generalize across spatial environments with similar structures and different visual features. This shows the ability for these cells to find abstract information about the spatial structure. This may provide evidence for schema-like learning, where the reuse of the spatial structure leads to efficient learning of new arenas.

There is also interest from the machine learning side in how schemas efficiently generalize previous knowledge to quickly learn new tasks [25]. With recurrent neural networks trained on learning tasks, Goudar et al. [26] found low dimensional manifolds that are representative of shared task variables. The low dimensional manifold could be considered as a neural substrate of the schema, which then limits connection weight changes when reused across problems.

Where do Schemas Live?

Many brain areas are involved with use of abstraction and decision making, such as the hippocampus, anterior cingulate, and prefrontal cortex. Particularly, the hippocampus is important for schema representations and the building of cognitive maps [27]. Tse et al. [22] also found that the hippocampus was important for the development of a schema. Zhou et al. [23] found neural representations in the orbital frontal cortex, a sub-region of the prefrontal cortex and important for learning associations to primary reinforcers (e.g. taste) and correcting reward related behavior [28]. The entorhinal cortex, which serves as a gateway of representations to and from the hippocampus [29, 30], has been shown to generalize across abstract tasks with similar statistical structures [31].

In decision making, there are often evaluations, where individuals judge their confidence and update their internal belief based on positive or negative feedback. Sarafyazd et al. [32] found that the anterior cingulate and dorsomedial frontal cortex were extremely important for switching behavior in an decision making task with uncertainty in the sensory information. With all of these areas in mind, it's important to understand the richness of each brain area and how they support cognition during difficult tasks.

1.2 The Role of the Hippocampus

The hippocampus [2] is an important brain region for schema formation and representation and it has been shown to be involved in numerous important cognitive traits: place cells [4], memory formations [7], theta oscillations [33], and abstract variables [34]. I now review some of the important features of the hippocampus in this section.

A good starting point is the finding of place cells in the Hippocampus [35], which are cells that encode the physical location in space. They have since been found in more general cases, such as mapping particular frequencies in a frequency spectrum [12]. As such, one could think of place cells as building blocks for cognitive maps [1]. Some hypothesize that place cells are more generally based on a formation of memory [4], where the hippocampus takes advantage of correlations to compress memory. This naturally expands to the idea that place cells, and more generally cognitive maps, apply to more than spatial mappings [11].

Place cells have also been hypothesized to be related to phase coding [36] in rodents, where cells fire in relation to the prominent theta (6-12 Hz) local field potential oscillation [33]. Importantly, the precession of firing relative to phase is independent of running speed [37] and direction [38]. It may be possible that the theta phase coupling is related to top-down mechanism for attention through rhythm coherence [39]. One critique is that non-rodents typically don't have large theta oscillations in hippocampus. However, Bush et al. [36] argue that this can be overcome with phase precession at arbitrary frequencies [40, 41], which was demonstrated in the bat hippocampus [42]. Further, theta reset occurs in both monkeys and humans [43, 44], which may be important for stimulus processing and memory formation.

In the processing of memory, abstract representations are extremely important for the generalization of variables [34] and cognitive flexibility. Abstract representations are related to the schema representations found in the hippocampus [3, 22, 23]. With this in mind, schemas may be important in memory consolidation in terms of finding efficient abstraction mechanisms.

From these results, there have been many hypotheses as to the role of the hippocampus [45–47]. Some believe in a declarative memory hypothesis [7], where it's believed that the hippocampus is important for consciously accessible memories but not memories that have been consolidated to neocortical sites. Others believe in a multiple trace theory [48], where episodic memories depend on the hippocampus with resilience due to repetition while some memories are stored elsewhere. While many different ideas exist, there isn't a precise definition of the functional role of the hippocampus [45].

No matter what the precise role is, the hippocampus is likely important in solving high dimensional credit assignment problems. Individuals likely need to form a memory representation to solve these tasks, and the hippocampus may play a role in finding abstract representations for efficient computations.

1.3 The Role of the Prefrontal Cortex

The prefrontal cortex (PFC) is a high-order area important for cognitive control and executive functions [49]. It can be broken down into many sub-regions with more specific cognitive functions [50]. Particularly, the motor cortex is involved in body movements and muscle activity [3]. The lateral PFC is involved with executive behavioral control, retention of information, selective attention, and behavioral planning [51]. The medial PFC, including the anterior cingulate, is involved in decision making, retrieval of long-term memory, learning associations, emotional response, and memory consolidation [52]. The orbito-frontal cortex is involved with representations of taste/odor reward values, learning stimulus reward associations, and reward related behavior [28]. From these regions, it is clear that the PFC has a high importance in the learning of complex decision-making tasks.

One of difficulties in solving decision making tasks is the evaluation of feedback related responses. In a reversal learning task, dorso-lateral PFC neurons predict choice preference switches [53], a result

consistent with the idea the the lateral PFC area represents a belief of the world. Further evidence for representing belief states is found in the WCST [54], where many dorso-lateral PFC neurons either encode success/failure or beliefs about the task. Belief states are represented in low dimensional spaces in the lateral PFC when learning spatial associations [55]. The lateral PFC also plays a role in working memory tasks [56, 57], where neurons are able to maintain activity for a short while after visual stimuli turn off. Lastly, the lateral PFC has been shown to be important in executive control of attention, where monkeys with lesions lost the ability to change top-down attention [58].

The medial PFC has been found to be extremely important in decision making tasks [59]. The anterior cingulate, a sub-region of the medial PFC, has been found to signal conflicting information [60, 61], a process useful for signaling changes in behavior. The anterior cingulate has also been found to be important in regards to remote memory [62]. The ventro-medial PFC has also been found to be important for consolidating declaritive memories [63]. Hoy et al. [64] found reward prediction error (RPE) neurons in the dorso-medial PFC along with the insula, which was found to communicate this RPE information to the dorso-medial PFC. The dorso-medial PFC has been shown to affect RPE signaling [65], which may be important in learning belief states. Domenech et al. showed the importance of the medial PFC during an exploitation vs exploration task [66]: they found that the ventro-medial PFC tracks reliability and predicts upcoming action outcomes while the dorso-medial PFC that guides behavior after responses to previous action outcomes.

The orbito-frontal cortex (OFC) plays a role in sensory integration and representation of rewards [67]. The reward value of odors, related to the satiation of the subject, has been represented in the OFC [68]. E.g., devaluing odors in an visual/odor association task leads to the result that the OFC and amygdala encode reward values associated with odors [69]. Zhou et al. found representations of schemas in the OFC [23], which may occur due the dependence of presented odors. Others have found dependence on many stimuli, including taste [70], somatosensory [71], and abstract representations of money [72].

From the associations of PFC in many tasks, The PFC likely is extremely important in solving high dimensional credit assignment problems. The OFC may be responsible for encoding sensory reward values. The medial PFC may integrate reward prediction errors, a process needed to consolidate beliefs during difficult tasks. The medial PFC could further be important for decision making and retrieval of belief states. The lateral PFC could be important for holding a working memory of stimuli as well as providing executive control of attention. With the link between the PFC and hippocampus [73], the PFC is clearly an important region when it comes to understanding neural mechanisms in solving complex decision making tasks.

1.4 Communication between Hippocampus and PFC

Both the hippocampus and PFC are important for understanding decision making and consolidation of memory [74] during high dimensional credit assignment problems. Communication between the two areas may be an important mode for memory consolidation [73]. Particularly, there has been evidence showing that the PFC and hippocampus couple via oscillatory synchrony, allowing bidirectional information flow [75]. While there are direct neural pathways between the two areas [76, 77], the PFC has also been shown to phase lock to the theta rhythm in the hippocampus [78]. Differences in synchrony between the hippocampus and PFC have been used to motivate the idea that associative learning may occur in the PFC while the hippocampus guides neocortical plasticity [8].

Connections between the hippocampus and PFC, particularly the medial PFC, appear during many aspects of tasks. Schlichting et al. [79] showed connectivity during rest periods, a process that enables the updating of memory. Minxha et al. [80] found an increase in synchrony during memory retrieval. Das et al. [81] found communication during memory encoding and recall; they found information flow from the hippocampus to the PFC from 0.5-8 Hz and the reverse direction from 12-30 Hz. Brincat et al. [8] found communication after feedback, particularly from the PFC to the hippocampus from 2-6 Hz after errors and from the hippocampus to the PFC from 9-16 Hz after correct trials. Bidirectional information flow may be important in decision making tasks, where information may signal updates to the memory.

1.5 High Dimensional Credit Assignment Problems

So how can high dimensional credit assignment problems help with the understanding, development, and use of schemas? Thus far, many studies focus on relatively straightforward tasks for abstractness, such as relating spatial locations to rewards [22, 24], sequences [23], or paired associations [82]. Others seek to generalize to more complicated tasks [83]. More complicated tasks, where credit assignment is needed and there are large numbers of objects, may reveal deeper characteristics of learning.

Consider the task of finding the underlying features that represent the ripeness of berries. Any feature, such as size, shape, amount of sun or water received, may be important. Every sample, or berry tasted, returns a single value that needs to be assigned to the collection of features and the assignment of credit to a single feature is difficult. This also involves efficiently storing information from all previous samples. There are two important ideas explored in these kind of tasks. First, how are schemas utilized to efficiently assign credit? Second, how are schemas utilized to efficiently store information? One can imagine non-schema like learning, where credit is assigned to every previous sample and all previous samples are remembered. This is inefficient and requires a tremendous amount

of memory and computations. It's likely that individuals will quickly forget previous samples, especially due to memory limitations [84, 85]. Then, individuals will learn extremely slowly or not at all.

To study this problem, a common framework is to switch the underlying "rule" after learning a task. In the berry foraging example, this would be akin to learning multiple berry species while developing a schema to efficiently identify important features. The question remains as to what schemas are utilized by individuals. If no schema is learned, then the individual will never efficiently solve the task.

One interesting question is the difference in schemas applied by different species. Direct comparisons have shown worse performance in monkeys than in humans in a variety of decision-making and memory-based tasks [54, 86–90]. With this in mind, one may assume that different schemas are found in different species, resulting in varying learning rates.

So what would a schema look like for high dimensional credit assignment problems? A schema should incorporate the knowledge that only one (or a few) feature(s) are important while incorporating a learning method that efficiently finds the primary features. While memory is not technically needed, as something like a win-stay lose-shift strategy (WSLS) [91] could solve these problems, memory of previous samples may converge on learned rules much faster. Efficient strategies have similar frameworks to partially observable Markov decision processes [92, 93] and in some cases a reinforcement learning [94]. Understanding the learned strategy can uncover the underlying schema.

1.6 Behavioral Task

To study schemas during high dimensional credit assignment tasks, I focus on data from monkeys and humans solving a variation of the Wisconsin Card Sorting Task [95]. This variant focuses on multi-dimensional aspects, and complex strategies are needed to solve the task effectively. The majority of this thesis is focused on subjects performing this task, and more details can be found in Sec. 2.1.

In this task, subjects see four images ("cards") on a screen, where each image contains a unique color, shape, and pattern (Fig. 2.1). In total, there are 12 features, and there are 64 combinations of three features. The subject is rewarded if they choose the card with the "correct" rule, a feature that is unknown to the subject. After a threshold number of correct trials in a row, the rule spontaneously switches without any cue. To maximize reward, the subject needs to learn a strategy that converges to the correct rule.

This task highlights the complexity of high dimensional credit assignment problems. The task requires generalization and understanding of feature importance rather than stimulus importance. That is, a naive strategy related to exact stimulus seen will not efficiently learn the rule. Further, this task highlights the problem of ambiguous credit assignment, where the subject needs to integrate information from multiple trials. That is, it is impossible for an ideal observer to figure out the rule from a single trial. At best

with a perfect strategy, one could find the rule after two trials. At worst with a perfect strategy, one could continually choose incorrectly and have to exhaust all 11 other features over the course of many trials. This highlights the need for cognitive flexibility, where the memory may constantly update based on the subjects behavioral state. One can imagine different internal beliefs when the subject initially searches, searches late in the rule block, tests features, or knows the rule.

One of the major difficulties is the large number of features. Given that both monkeys [96] and humans [84, 85] are memory-limited to about 3-5 objects, they may not be able to hold the 64 stimuli in memory. Assuming that the subjects strategy holds one value per feature, it is unlikely for subjects to remember values for all 12 features. To solve this, efficient coding and memory consolidation is needed.

Lastly, and possibly most importantly, rule switches demand cognitive flexibility and schema formation. It may take many trials to gradually learn the rule on the first block. After developing a schema, rules in future blocks will be learned faster, leading to more reward overall. In a sense, the subject needs to "learn to learn," where they need to learn an efficient learning strategy for learning the rule.

To receive large amounts of reward, the subject must form a schema representation. E.g. a subject could bias their choices towards features that were recently rewarded. Memories could be consolidated by removing information about non-chosen features or features that are likely unrewarded. The schema could be the reason for generalization across features, cognitive flexibility, and consolidating memories to a small set of possibilities.

1.7 Conclusions

To study the neural representations necessary for individuals solving high dimensional credit assignment problems, I focus on the WCST dataset from the Buffalo lab. This task exemplifies cognitive flexibility, generalization, schema formation, ambiguous credit assignment, and memory consolidation. Behavioral modeling is needed to understand how the subjects learn to solve this task. Only then can one find neural correlates to the underlying belief state. And finally, one can learn about the representations and use of schemas in these complex tasks.

There are many hurdles to overcome when focusing on this dataset. E.g., the subject is allowed naturalistic free viewing behavior during the choice period, where neural activity may not be trial-aligned. To compensate, neural data can be directly aligned to the eye movements. Another major difficulty is understanding the subjects' beliefs. If, for example, the subject chose a cyan striped triangle, did the subject choose the card because it was cyan, striped, a triangle, or some combination of the three features? The experimenters may have true knowledge of the world, but the subject has a limited view of the world. To find neural correlates, one first needs to understand the belief of the subject.

To solve the belief state problem, one may assume a "simple" behavioral model where the subject

focuses on the rule towards the end of a rule block. However, not only does this remove many interesting trials, it also assumes a belief state of the subject that may not be true. As shown in chapter 3, the subject can be distracted by other features during the criteria trials, which means that the subject does not always focus on the rule. This provides rational for behavioral modeling, where all trials can be used.

The problem is confounded further by the sheer number of features compared to the number of trials in each session. Given that the subject takes 30-40 trials to find the rule per rule block and there are on average around 1000 trials per session, each rule may only occur a few times. In some sessions, due to randomness of the rule switch, only 11 out of 12 features are dictated as the rule. In most sessions, there is at least one feature where the rule is that feature once. With this in mind, validating the subjects belief is extremely difficult, since validation requires that the feature should be the rule multiple times throughout the session. Behavioral modeling can remedy this problem, since beliefs of all features occur sporadically throughout all trials.

There are also difficulties related to the neural data. The data involves multi-area, simultaneous recordings across months of sessions. With this in mind, one needs to aggregate statistics across all sessions to find simple but meaningful results. Further, precise locations of electrodes and spikes are not known. While there has been a great deal of effort[1] in localizing the placements, one cannot precisely know the locations until an autopsy is performed. For example, the implanted electrodes may bend depending on scar tissue. Throughout this thesis, I use general approximations courtesy of Mike Jutras.

With these difficulties in mind, I now highlight the main findings within this thesis. First and foremost, from the behavioral modeling (Chapter 4), monkeys perform this task with a reinforcement learning framework while humans perform like a memory limited ideal observer. Monkeys also pay particular attention to the previous card, where they preferentially choose or avoid the previous card if it was correct or incorrect. These strategies highlight potential schemas of the subjects, where they take advantage of the task structure by biasing their choices towards recently correct features. More details on the relation to schemas can be found in the discussion (Chapter 9). Second, there is evidence for the behavioral models within the neural data. That is, there are neural correlates related to when the subject focuses on particular features. There is a particular prevalence of feature belief during the feedback period in the PFC, hippocampus, and basal ganglia, showing the importance of the feedback period in computing belief updates.

In a related analysis, there are differences in the LFP amplitudes and timings during the feedback period in the hippocampus and PFC; differences occur depending on the feedback and/or previous feedback. There is also evidence for communication between the hippocampus and PFC in the LFP, similar to results found by Brincat et al. [8]. In the 4-5 Hz band, information flows towards the HPC, an effect that is stronger during incorrect trials. In the 20-24 Hz band, information flows towards the PFC,

[1] Special thanks to Mike Jutras for all of the hard work in electrode localization

an effect that is stronger during correct trials. Interestingly, there is a reversal of flow during incorrect trials in the 4-5 Hz band, where information flows towards the PFC after a few hundred milliseconds.

Throughout both the behavioral modeling and feedback analysis, the activities timing shows interesting behavior. There are different processes that occur before and after ~ 400 ms after feedback begins. The early feedback signal may be important for quick belief evaluations while the late feedback may be related to further computations in belief updates. See Chapter 9 for more details and interpretations.

The research here also involved the development of the Non-Stationary Dynamical Mode Decomposition (NS-DMD) method, which generalizes DMD methods to work on non-stationary data. This allows for further comparisons during the feedback period, where NS-DMD results agree with results found from simpler methods. The benefit is that the NS-DMD approach is a dynamical systems approach, where models approximate the full dataset. Its preliminary use during the feedback period shows promise for further analysis. Further, collaborators have already begun using the method in other areas, such as in spiking data, finding general phases of modes, and theoretical coupling between multiple oscillators.

The layout of the thesis is as follows. Chapter 2 focuses on the methods, diving into task details, details related to the origins of spiking and LFP data, typical time-frequency analysis, and general processing of the data. Methods specific to each analysis are included in subsequent sections. Chapter 3 focuses on introductory data analysis, beginning with an analysis of near-ideal simulated behavior. Then, general subject behavior is explored, including a visual search strategy during the selection period. This also includes an initial search of neural data using spectrograms, slow timescale power fluctuations, and demixed principal component analysis. In Chapter 4, the subjects choices are modeled with various behavioral models. The neural data is then correlated to the behavioral models in Chapter 5, finding belief specific computations. Feedback related analyses are presented in Chapter 6, where there are different responses and information flow depending on if the trials were correct or incorrect. Chapter 7 focuses on additional analyses, beginning with an attempt at rule decoding assuming a rule belief during criterion trials. This chapter then analyzes phase reset, eye movement correlates, and decoding when the subject is fixating on their final choice compared to initial card viewings. Chapter 8 introduces the NS-DMD method. Finally, the discussion (Chapter 9) wraps up the thesis, highlighting results and interpretations of the roles of each area and the use of schemas during these tasks.

Chapter 2

Materials and Methods

This section includes a broad overview of the main materials and methods included in the bulk of the thesis. This includes the datasets, how they were collected, pre-processing, and general processing required before more in depth analyses. Specific methods involved in particular analyses are introduced in their respective chapters.

2.1 Wisconsin Card Sorting Test (WCST)

Subjects perform a variation of the Wisconsin Card Sorting Test (WCST) [95], a task typically used in the clinical setting to measure frontal lobe dysfunction [97]. This is a complex decision making task requiring the subjects to perform abstract reasoning and problem solving; it also requires the use of visual processing, attention, and working memory. It also holds the potential to understand the dynamics of learning. In the beginning, the subjects may have an inefficient representation of the task, but over time their representation is more abstract, can generalize better, and ultimately solve the task to receive reward faster. Thus, this task is a perfect candidate of analyzing how the brain learns, makes decisions, forms abstract representations, and holds information in memory.

The WCST was performed by many subjects across the Buffalo, Knight, and Lin labs with varying amount of neural data recorded. Generally, I focus on data from monkey S, that has a plethora of both neural and behavioral data. For the behavioral analysis, I focus on all behavioral data from 4 monkeys and 10 10 humans.

Task Structure

This variation of the WCST is designed as follows: the subject fixates on a cross in the middle of the screen to initiate each trial. After fixating for a fixed time, the cross disappears, and four objects appear on the screen. Each object consists of a different color, shape, and pattern. The subject must choose

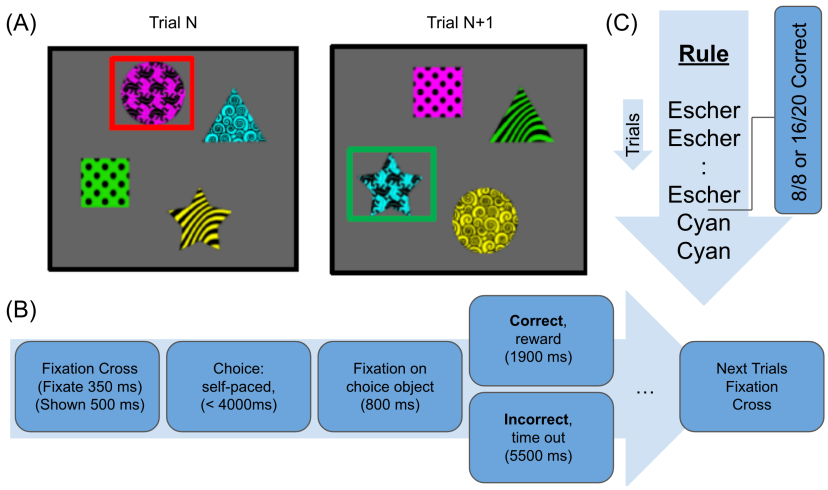

Figure 2.1: Schematic of the Wisconsin Card Sorting Test for the NHPs. There are different stimuli, timings, and trials to criteria for the humans (see text). The task structure is the same for both species. (A) shows two sample images of what the subjects sees on consecutive trials. The subject chooses one of the four objects and either receives a reward or timeout depending on if they were correct or incorrect. The red and green boxes indicate sample choices corresponding to an incorrect and correct choice. In this case, the correct choice is object with the Escher pattern. The timeline of the task is shown in (B). (C) shows the a depiction of the underlying rule for consecutive trials. Once the subject chooses 8 out of 8 or 16 out of 20 correct trials in a row, the rule switches from (in this case) Escher to cyan.

one of the four objects based on an individual feature, known as the rule, which is unknown to the subjects. For example, the subject may receive positive feedback when choosing the purple card and receive negative feedback when choosing a non-purple card. After reaching a threshold of trials correct, the governing rule will spontaneously change to something else. The rule change is not cued. If the rule switches dimensions, i.e. from color to shape, the switch is termed extra-dimensional. If the rule stays in the same dimension, i.e. from purple to green, the rule switch is intra-dimensional. See Fig. 2.1 for a schematic of the task.

Minor Differences in Species

There are numerous differences in the task design. Some design choices were intentional and are necessary for completion of the task by different species. Others are more dramatic and may cause differences in strategies.

First, the NHPs are required to fixate on the fixation cross for 350 ms before the trial begins, and the cross is shown for an extra 150 ms. The objects appear on the screen 98 ms after this. Humans are required to fixate for 300 ms, and the objects appear immediately afterwards. Second, NHPs have 4000 ms to make a choice, which is done via eye-tracking: a choice is defined when the subject's gaze lies within a bounding box around an object for 800 ms. Humans have 4000 ms to make a choice, done via an arrow key button press. After 98 ms for NHPs, the objects disappear from the screen, and feedback begins. Feedback begins instantly for humans. Lastly, NHPs receive a juice/chow mixture and hears a

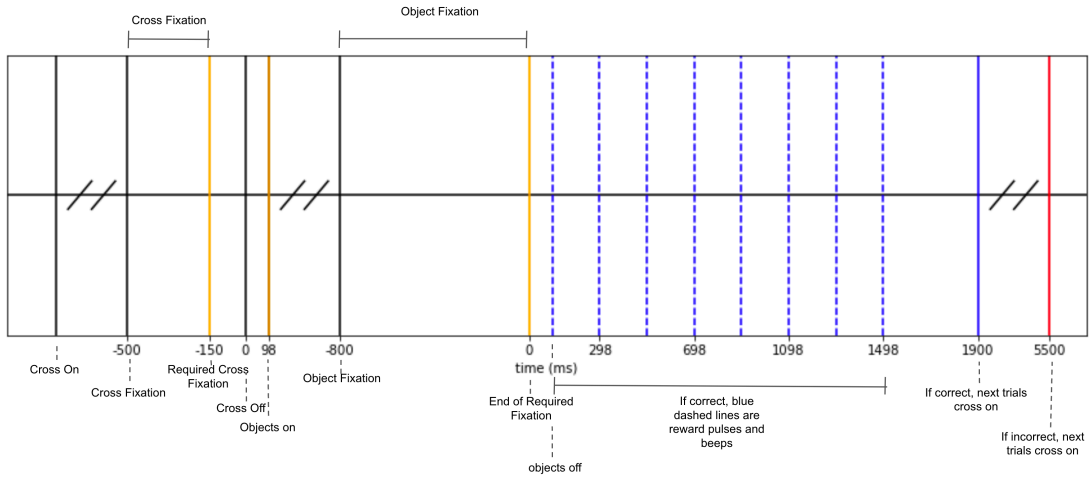

Figure 2.2: Timeline of events for non-human primates. Diagonal lines indicate variable lengths of time

beep 8 times every 200 ms for a total of 1400 ms when correct, and they receive a timeout where nothing appears on the screen when incorrect. The fixation cross on the next trial appears 1900 ms or 5500 ms after the end of object fixation on correct and incorrect trials respectively. Humans, receive visual feedback, with a label saying either ”correct” or ”incorrect.” The next trial for humans begins after 1000 ms. Specific times for NHPs are indicated in Fig. 2.2.

A quick note on the values ”98,” ”1900,” and ”5500” above: these were analyzed after the recordings were done. The inter-trial intervals 1900 and 5500 were found by averaging the inter-trial difference. The 98 ms delay has a subtle origin. From the design of the task, object appear/disappear when the command GFLUSH occurs. Unfortunately, this command was not documented. By taking trials where the trial timed out (i.e. the monkey did not choose an object in time), one can look at the time difference between the nearest recorded commands. The average time is 98 ms.

For three out of four NHPs, the rule switches after either 8 correct trials in a row or 16 out of 20 correct trials in a row on the current rule. For one of the NHPs, Monkey C, there was a glitch in the programming, which was attempted to be fixed by modifying the trial switch criterion. Ultimately, this ended with the majority of rules switching after either 8 correct trials in a row or 15 out of 20 correct trials in a row on the current rule. After further inspection, there are numerous rules that switch anywhere from 7 to 16 correct trials. For humans recorded by the Knight lab, the rule switches after either 5 correct trials in a row or 8 out of 10 correct trials in a row on the current rule. In the Lin lab, there was an error in programming, and the rule could switch by reaching the trials to criterion without regard to the current rule. E.g., if the subject was correct for 6 trials in a row, the rule would switch both after 5 trials and on the trial afterward. This may impact the strategies of these subjects.

Lastly, for NHPs, every now and then the experimenter will introduce a short break of about 12 minutes where nothing is shown on screen. This can lead to a resetting of internal beliefs, although this

only occurs a few times throughout each session.

Major Differences in Species

There are a couple of major differences in the task design. First, the proportion of intra- vs extra-dimensional shifts are different. In the human version, the rule changes to any of the 11 other features randomly. The monkey version was designed such that there is an equal probability of intra vs extra dimensional shifts. This, in principle, could lead to different strategies since an ideal agent would bias their strategy towards choosing intra-dimensional features after a rule switch. However, we don't believe that NHPs take advantage of this fact, as shown in Sec. 4.7.2.

The second difference is the randomness of the cards that appear. Every card is randomly chosen on every trial for NHPs. However, there is a set sequence of cards that the humans see. The rules and rule changes are still completely random. This, in principle, could be memorized by human subjects. This seems unlikely as remembering 4×300 cards would be hard and not productive given that the relevant rules are still random. At best, it could help the first few trials, but it would be unlikely to change any result after a few trials.

Lastly, the most important difference is the instruction. NHPs learn solely by trial and error. Humans receive instructions on the screen before the task begins. There are two groups of humans that receive different instructions. The first group reads: "In this experiment, you will see 4 cards on each trial. Each card has 3 unique features (color, shape and texture). The card containing the correct feature will be the correct choice. The correct feature might change during the task. Press any key to continue. The answer is given by pressing one of the four arrow keys that corresponds with the selected card position on the screen (up, down, left or right). You have 4 sec to provide the answer, or the trial times out. Good luck and thank you!" The second group reads: "In this experiment, you will use one of the four arrow keys on each trial as a response. The 'correct' or 'incorrect' feedback will be provided following each choice. Your task is to maximize the number of correct responses." The key difference in the two groups is the amount of instruction. The first group is told how the task is done. The second group has a minimal amount of instruction, which is much more similar to the NHPs.

Modifications to Task For Monkey S

For Monkey S, the main subject for analysis, there are a couple modifications and bugs. Some sessions have a bug whereby rule blocks can last for a single trial, which can lead to a rule occurring effectively twice in a row. Next, from 2018-10-15 to 2018-12-07 and 2019-01-30 to 2019-02-01, a bug occurred in which the timeout period was significantly shortened. In these sessions, the timeout period was only 1480 ms. Lastly, between dates 2018-12-10 to 2019-01-25, different stimuli were shown to Monkey S. Instead of the stimuli shown in Fig. 2.1, different colors, shapes, and patterns were shown. For the majority of

analysis, I focus on sessions before the first timeout period shift.

Behavioral Pre-processing

The first and sometimes second rule is typically biased. For this reason, the first two rule blocks are dropped for all subjects. The subjects, especially the NHPs, get tired by the end of the session, so the last rule block is dropped. Naturally, any trial where the subject does not make a choice is dropped. Sessions with bugs, such as single trial rule blocks, are dropped.

2.2 Experimental Details

The four adult female rhesus monkeys (Macaca Mulatta) are labeled as Monkeys S, C, B, and T. Their age at the start of the recordings was 9.1, 10.8, 19.9, and 10.3 years respectively. Their weights at the start of the recordings was 7.9, 7.6, 8.5, 5.9 kg respectively. The subjects completed 42, 29, 89, and 23 sessions respectively, with 959 $\pm$ 269 trials per session. Evaluated by the experimenter, every 31 $\pm$ 5 minutes, the subjects rested for 12 $\pm$ 3 minutes while seeing a blank screen. Programming occurred with NIMH Cortext or NIMH MonkeyLogic software. All subjects were equipped with eye tracking (EyeLink 1000 Plus, SR Research), and all subjects were head-fixed with a titanium rod. Monkeys S and C had microdrives implanted for corresponding neural analysis. All procedures were carried out in accordance with the National Institutes of Health guidelines and were approved by the University of Washington Institutional Animal Care and Use Committee.

Each NHP was head-fixed in a dimly illuminated room, positioned 60 cm away from a 19-inch CRT monitor with a resolution of 800 $\times$ 600 pixels, corresponding to 33 degrees by 25 degrees of subtended visual angle.

There were no neural recordings for the humans. The studies involving these human participants were reviewed and approved by the Institutional Review Boards of University of California, Berkeley. All participants provided their written informed consent to participate in this study and received a small compensation. While not studied here, data does exist from a group of human participants with neural implants.

2.3 Origins of Neural Data

It's important to understand the source of the the data. Ultimately, one hopes to discover the connection between firing of neurons and the thoughts and behaviors of the subject. This section highlights characteristics and origins of the local field potential (LFP).

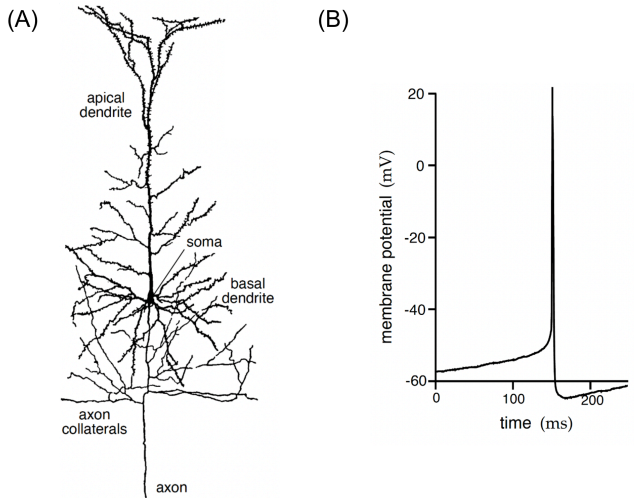

Figure 2.3: (A) Depiction of a typical cortical pyramidal neuron cell (Dayan & Abbot Fig. 1.1 [4]). Axons typically travel a large distance: large fractions of the brain, or even the whole body. (B) Action potential (spike) of intracellularly recorded cultured rat neocortical pyramidal cell (Dayan & Abbot Fig. 1.2 [4]). The fluctuation in potential lasts for around 1ms, and typically there is a refractory period where the neuron is unable to spike for a few ms.

An electric probe, or electrode, is placed inside the brain. The probe measures the electric potential V, referenced to a ground that is typically placed outside the brain, on the ears, or somewhere respectively far away from the electrode. Any fluctuations in potential are measured, which can range from brain activity to artifacts such as wall noise. Even eye movements and muscle artifacts are known to influence the measurement from the probe [98]. Removing artifacts, or activity not associated with the brain, will be discussed in Sec. 2.4.

So what brain activity is measured? When a neuron fires (Fig. 2.3(A)), there are a plethora of moving charges. Dendrites receive inputs from other neurons, which causes sodium ions to flow into the neuron. If enough sodium enters, causing a depolarization from the resting potential, a positive feedback process begins where more sodium ions enter. The potential of an example cell is shown in Fig. 2.3(B). The ions "spread out" due to coulomb forces throughout the neuron, which leads to more sodium ions entering, potassium ions leaving the neuron, and later positive calcium ions flowing into the neuron at the axon terminals [4]. The influx of calcium is responsible for sending neurotransmitters to nearby neural dendrites, restarting this cycle. There is a refractory period of a few ms, where the ions return to balance out the potential. There is a large literature building on the seminal work by Hodgkin and Huxley [99] modeling the ion currents. Since there are different places where ions flow in and out of the cell, there is a formation of areas of relatively large and relatively small potential, known as potential sources and potential sinks [100]. Then, there are return currents as the system is restored to rest. From this, neuronal firing is typically characterized as collections of changing dipoles [100]. Given the complicated charge distribution and ever evolving dipoles, one could in principle use Coulomb's law to find the potential everywhere else in the brain.

A problem arises due to the sheer number of neurons. The potential at the electrode is a sum of every potential contribution from every neuron. If the neurons fire randomly compared to each other, the relative contributions at the electrode can cancel out or look like noise. Then how exactly can a measurement reveal anything? The secret is that neurons fire somewhat synchronously [101]. Specifically, post-synaptic transmembrane currents in the dendrites form an extracellular sink. An extracellular source is needed elsewhere, which occurs downstream on the neuron. The extracellular currents, deemed return currents, between sinks and sources are relatively synchronous between nearby neurons due to the relatively slow timescale [100]. If a collection of neurons all fire at a similar time, there is a relatively large contribution to the potential at the electrode compared to the random addition of all of potentials from all other neurons. Also, the contribution from far away neurons, which may not be synchronous, dies like $1/r^2$.

Another important point is that neurons typically fire multiple times [4]. If a collection of neurons all fire simultaneously, there is an associated increase in power in the LFP at the frequency at which the neurons fire [102].

Hypothetically, if one knew every current from every neuron, one could accurately find the potential at the electrode, but this impossible in practice. Also, what's usually of interest is the inverse problem, discovering information about the neurons from the LFP. This is practically impossible due to a plethora of charge distributions that can all lead to the same potential.

Geometry also plays an important role in the potential measured. Since the neurons form dipoles, there is an associated directionality in potential variation. In thalamocortical cells for example, dendrites emerge relatively equally in all directions, leading to a field that naturally cancels itself out [103]. Fortunately, the cortex is a layered structure, with a planar organization filled with pyramidal cells whose processes are oriented orthogonal to the layers, naturally forming a linear like shape with a spatially separated source and sink. In the cortex, pyramidal cells typically all face the same direction in a planar manifold. However, the plane then folds on top of itself multiple times. This complicates things further since the potential sinks and wells can now form dipoles between the folds of the plane [100].

To summarize, the potential is typically thought of as a measurement of synchronously firing neurons. More specifically, measurements that are not noisy tend to arise from large collections of synchronously firing neurons. With this in mind, it's also important to think of what behavioral quantities are readily measured.

Typical Behavioral Correlates

Typically, the frequency spectrum of the LFP ranges from 1 to 100 Hz [100] to sub-500 Hz [102]. Power in the 100 to 500 Hz range could occur when neurons fire every 1 to 10 ms, which is on the order of the refractory period mentioned in [4]. The LFP spectrum is typically divided into different ranges, although

the exact ranges depends on species and by the experimenter. Shi [104] summarizes the ranges as follows for non-human primates. The delta range is from 1 to 4 Hz. This typically is associated with slow wave sleep [105]. Next, the theta range is from 5-8 Hz, and is commonly observed in studies of learning, memory [106, 107], and spatial navigation [1]. This also seems to be an extremely important range in the mouse brain, where different processes synchronize to the theta frequency in a frequency coupling process [33]. The alpha range is from 9 to 14 Hz, which typically is thought of as underlying network coordination [108] and attention [109]. The beta range is from 15 to 30 Hz, and is related to motor processes [110] and working memory [111]. For example, [112] highlights that isometric contraction of fingers, wrists, shoulders, feet, and tongues have been found to relate to a drop in beta power. The last range, of particular importance, is the gamma range from 30 to 200 Hz. Literature sometimes divides this into low gamma from 30 to 50 Hz, high gamma from 50 to 100 Hz, and very high gamma above that. Low and high gamma are related to cognition [113] and in frequency coupling with slower bands [114]. Very high gamma has been shown to correlate to spiking [115].

Given this association of different frequency ranges with different behavioral correlates, it's important to think about the frequencies of artifacts. For example, 60 Hz wall noise causes a problem when measuring high gamma. Eye movements in non-human primates typically occur every 200 ms, corresponding to 5 Hz. Thus, an increased power at 5 Hz could be due to synchronized firing of neurons or some computation related to visual processing. This is complicated further by eye movements being reported as causing artifacts in LFP [98] at 5 Hz.

Muscle artifacts have been shown to cause problems at a wide range of frequencies [98]. The peak frequency of the masseter muscle (chewing related) occurs at 50-60 Hz. Frontalis (wrinkling of the brow) occurs at 30-40 Hz. The lower band limit of the activity of these muscles is about 15 Hz while the high end activity is well above 100 Hz. Muscles that control saccadic eye movements produce activity that peaks at 65 Hz. Representation of sounds in the auditory and somatosensory cortices is observed between 75-110 Hz. Clearly, there is a wide range of non-cognitive processes that are visible and need to be removed.

It's important to have pre-processing steps that take into account both artifacts and the large range of possible frequency ranges reported. In Sec. 2.4, I discuss localizing the LFP signal in an effort to remove artifacts, relying on the fact that artifacts typically spread much faster than cognitive processes. In Sec. 2.7, I discuss a method for finding frequency ranges in a data driven way, without regard to previous ranges.

Frequency Power Law

The spectrum of the LFP is known to decay like a power law [116]. This affects the pre-processing and dimensionality techniques like principal component analysis, which will preferentially select components

of the LFP at lower, and therefore higher power, frequencies. This section focuses on possible reasons for this decay.

The first common reason for the power law decay occurs from dendritic low pass filtering ([100, 102]. A filtering effect occurs around single neurons, which could be explained by a slight shift in timing between potentials from the soma compared to the synapse [117]. This naturally impacts the summation of potentials from different neurons, where power is larger at lower frequencies.

The other common reason occurs from the naturally large timescale of lower frequencies. At lower frequencies, there is a larger window of time in which neurons can fire and impact the power in the same fashion. For example, let's imagine two neurons fire with a constant time delay of 5 ms. If the frequency of interest corresponds to a wavelength of 30 ms, then the two neurons will both fire near the peak/trough, which sums to increase the total power of the frequency. If, however, the frequency of interest corresponds to a wavelength of 10 ms, then one neuron could fire at the peak and the other could fire at the trough. This leads to a cancellation of the power at this frequency. This thought process shows that more neurons can fire where the voltage constantly sums together without cancelling. At larger frequencies, the potentials can cancel out, leading to a smaller power at higher frequencies [100].

LFP and its Connection to Spikes

If the LFP is measured at a high enough sampling rate, it is possible to find individual spikes in the data. For example, Beth's lab measures at 30 KHz, which is a high enough resolution to see individual spikes if the electrode is close to neurons. The potential is known to spike with a fluctuation on the order of ~ 100 mV (Fig. 2.3) at a timescale of about 1 ms [4]. Spikes are typically found by hand or by taking advantage of more recent unsupervised techniques [118]. After finding them, spikes are a great source for finding correlates to behavior due to the very high spatial and temporal resolution.

As mentioned before, studies have shown that a correlation of the LFP signal with spikes can occur for the very high gamma range of about 100 to 200 Hz. However, another important finding is the influence of spiking artifacts in this range. That is, [119] has shown that spikes can easily affect the LFP even down to 10 Hz. One needs to be careful to say that their correlation is truly due to a neurological effect and not just an artifact of spiking.

Communication Through Coherence

There has been a lot of interesting discussion on whether the LFP is important to the brain; is the LFP necessary to computations? One could argue that anything can be computed with the sheer number of neurons and connections, much like a hypothetical large computer. On the other hand the LFP affects the surrounding potential of every neuron; therefore it could affect when neurons fire. One may even

Technique	Source	Temporal Resolution (s)	Spatial Resolution (mm)
Single Unit	Mitra [5]	10^{-4}	10^{-2}
LFP	Dubey [121]	-	$(3 \text{ to } 30) \times 10^{-1}$
LFP	Me	-	$< 4 \times 10^{0}$
EEG/MEG	Mitra [5]	10^{-3}	10^{1}
EEG/MEG	Glover [122]	10^{-3}	$(1 \text{ to } 2) \times 10^{1}$
MEG	Buzaki [100]	10^{-3}	$(2 \text{ to } 3) \times 10^{0}$
fMRI	Mitra [5]	10^{0}	10^{0}
fMRI	Glover [122]	$(0.1 \text{ to } 3) \times 10^{0}$	$(0.5 \text{ to } 4) \times 10^{0}$

Table 2.1: Spatial and temporal resolution for different techniques. I approximated the spatial resolution in Fig. 2.5, defined by when the coherence decays to 0; the majority of coherence is less than 1 mm. The relevant neural scales are dendrites (10^{-3} mm), neurons (10^{-2} mm), cortical layer (10^{-1} mm), cortical column (1 mm), and map (10 mm) [5].

argue an evolutionary approach; there are possible mechanisms of the LFP that could be beneficial, and evolution tends to take advantage of mechanisms that can be taken advantage of. Specifically, mutations can lead to different possibilities, and beneficial computations lead to more survival.

One possible computational mechanism that the LFP could support is communication through coherence [39]. Forward communication, e.g. visual information leading to more complex cognition further in the brain, is easily explained with neural connections. Backward communication, such as attention to visual vs auditory stimuli, could be explained by speeding up frequencies in combination with inhibition [39]. There is low and high frequency coupling, such as from theta and gamma [114]; if the amplitude of low frequencies increases slightly, then neural firing in one area of the brain could occur earlier than others. If these neurons fire earlier, then the gamma power increases and other areas are inhibited. This could serve as a mechanism to direct attention to different external stimuli. There is evidence for this [39], for example, a finding that attention to specific stimuli in V1 always propagates to V4 [120], a downstream area.

Relevant Scales and Real Numbers

The relevant scale of the LFP and the estimated number of neurons contributing to it can help inform us on the likely observed information. There are pros and cons with each recording method, and some information may be better tackled with spikes, EEG, MEG, or fMRI. Tab. 2.1 gives typical scales from different sources. The ideal method is one with a very high temporal and spatial resolution. While achieved with single unit activity or spikes, this requires invasive recordings and relatively few neurons in the brain are recorded. More area coverage is desirable, which can be accomplished with EEG, MEG, fMRI, and other imaging methods. However, one may also desire a finer resolution to discover population trends, such as with LFP.

One question is how many neurons contribute to large effects in the LFP. Typical spatial resolutions of the LFP are given in Tab. 2.1. Using a spatial resolution of 3 to 30×0^{-1} mm and a average macaque

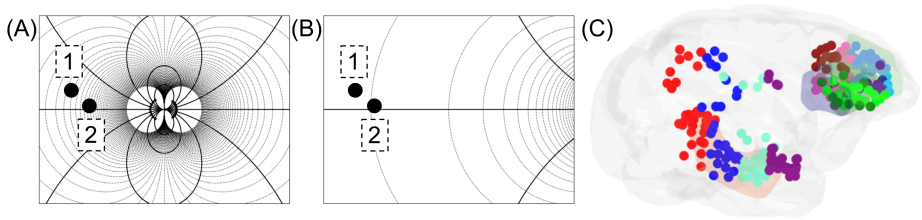

Figure 2.4: Schematic of the impact of a hypothetical potential field from a "close" source (A) and a "far away" source (B) on two electrodes labeled "1" and "2". The same electric field flux lines (solid) and equipotential lines (dashed) are included in (A) and (B). In this schematic, the potential difference between electrodes "1" and "2" is larger when the source is "close" vs "far away." As the distance increases, the spatial gradient between both electrodes approaches 0. (C) The groups of electrodes for Monkey S's recording on date 2018-08-02, where colors separate grounds and manufacturing company. Plots (A) and (B) are generated with code from https://github.com/tomduck/electrostatics/tree/master.

monkey brain size with approximately 1.4 billion neurons [123] and a 80 cm^3 volume [124] (averaged between males and females), the number of neurons that regularly contribute to the LFP is around 500 to 500,000.

2.4 Localization of LFP

The local field potential is very widespread. One common question is, "how local is local?" If more neurons are synchronous, the signal could propagate further. Different frequencies can also propagate over different distances. If the extracellular medium is capacitive, low frequencies should travel farther than high frequencies [121]. If the medium is resistive, all frequencies should travel the same amount. Dubey et al. [121] found, however, a bandpassing effect where frequencies between 60 and 150 Hz traveled approximately 1.2 to 1.5 times further than other frequencies.

Another issue with locality is the presence of noise. Noise from the ground signal or from muscle artifacts can show up in measurements. It's important to remove, as best as possible, artifacts and "non-local" signals. There are two common ways to do so: with ICA and bipole subtraction. The underlying assumption in both methods is that non-local signal typically has a somewhat equal weighting across all electrodes. That is, an artifactual signal will propagate much faster than signal passed through neural connections, and the artifactual signal will affect all electrodes simultaneously. Both methods are discussed, and most of the analysis in this thesis uses the bipole subtraction method.

Bipole Subtraction

Bipole subtraction to localize the signal relies on subtraction of two nearby LFP signals, which finds the local spatial derivative [125]. A schematic is shown in 2.4, where there are two sample sources: one "close" (Fig. 2.4(A)) and one "far away" (2.4(B)). The goal with bipole subtraction is to keep the close

signal while removing the far away signal. Both the close and far signals are present in each electrode. When the signals are subtracted to form a "bipole," the "close" signal will have a larger amplitude than the "far" signal due to the spatial derivative decreasing as a function of the distance from the source. Mathematically, this can be seen from Coulomb's Law by estimating a dipole source:

$$\phi(r) = \frac{q\cos(\theta)}{4\pi\epsilon_0 r^2}, \tag{2.1}$$

where q is the charge and θ is the angle between the dipole moment and the position vector. Computing the difference in potentials at a point R and a point $R + d$ reveals, up to a constant c, a difference that decays with distance:

$$\phi(R+d) - \phi(R) = c\left[\frac{1}{(R+d)^2} - \frac{1}{R^2}\right] = c\left[\frac{d^2 - 2Rd}{R^2(R+d)^2}\right] \sim \frac{1}{R^3}. \tag{2.2}$$

Thus, subtracting two signals removes "far" sources and maintains "close." A smaller electrode distance (d) leads to a more local signal.

There are two considerations when subtracting signals. First, the grounds must be the same. Otherwise, any artifactual signal that independently affects the grounds will remain in the subtracted signal. Second, the intrinsic impedance of each electrode tip should be the same; otherwise, non-local signal will not be subtracted away. In the Buffalo lab, the locations of each ground were recorded as well as the manufacturers of each electrode. Unfortunately, measurements of the intrinsic impedance are difficult *in vivo*, and the measurements were not recorded before placement into the brain.

With this in mind, I bipole subtract independently for every session with the following steps. Every electrode was grouped by their respective manufacturer and ground source (e.g. Fig. 2.4(C)). Second, for every group, the electrodes were paired into bipoles based on minimizing the distance between each bipole in a greedy fashion. The closest two electrodes were paired, followed by the next two closest, etc. until all electrodes were paired. I drop any bipole with an inter-electrode distance greater than 3 mm. The location of each bipole is halway between each individual electrode. This process ensures that the grounds are the same, the intrinsic impedance of each electrode is most likely similar, and the distance between electrodes is relatively small. This leads to a relatively local signal for each bipole, as shown in Fig. 2.5.

ICA Artifact Rejection

Like with bipole subtraction, this method assumes that non-local signal affects all electrodes. Then, artifact rejection can be thought of as an Independent Component Analysis (ICA) like problem [126]. The classical example in ICA analysis is the cocktail party, where there are n conversations and m

microphones. Every microphone picks up each conversation at some variable amplitude. ICA assumes independence of each signal (signal x does not influence signal y in any way), and solves the inverse problem of finding each conversation. A similar thought experiment can be performed with noise and electrodes: there are n sources of noise and m electrodes measuring signal plus noise. Solving the inverse problem leads to a number of independent components. If any component has a widespread influence, or the weights across the brain are all non-zero, then this likely is a non-local component. These components are subtracted from the signal for each electrode [127, 128], leaving only the local, non-noisy signal.

There are numerous difficulties with this method. In the standard example, there are a known number of sources. In the real brain example, there are an unknown number of sources of signal and noise. How many signals should one decompose the data into? 1 mode is clearly too few, but it's unclear how many more sources exist. One method of solving this is to run ICA independently with different numbers of modes. If a sensible number is chosen, then the larger number iteration should contain the same modes and weight matrices as the smaller number iteration plus extra modes. Due to random initialization, the results should be consistent. One could increase the number of modes in ICA until the new modes drastically differ than the previous iterations modes. This does not a guarantee the "correct" number of modes.

Another possible issue is that the amplitude of the signal decays with increasing frequency. Different sources of noise and non-local signal could appear in different frequency bands. ICA may find sources at lower frequencies without finding sources at higher frequencies due to the decaying amplitude. This is complicated further considering that a standard trick to speeding up the computation of ICA is to first reduce the number of dimensions with Principal Component Analysis. One solution is to bandpass the data into different frequency bands before applying ICA.

Verification of Locality

After bipole subtraction or ICA artifact rejection, locality is verified with a method from Lachaux et al. [125] for visualizing the spatial extent of the LFP. They suggest calculating the coherence between all bipoles across minutes to hours. While electrodes are expected to be coherent on short time scales, it is unlikely for two electrodes to be coherent for a long timescale unless they are measuring similar quantities.

I compute the coherence between every pair of raw electrodes and pair of bipoles in Fig. 2.5. The bipole coherence is smaller than the raw electrode coherence, indicating that the bipole subtraction procedure localized the signal. The bipole subtracted signal has a cutoff range of about 4 mm depending on the frequency of interest. Lower frequencies are more coherent than higher frequencies. Note that the coherence in theory should be 1 at a distance of 0 mm. This means that the local signal likely is less than 1mm away.

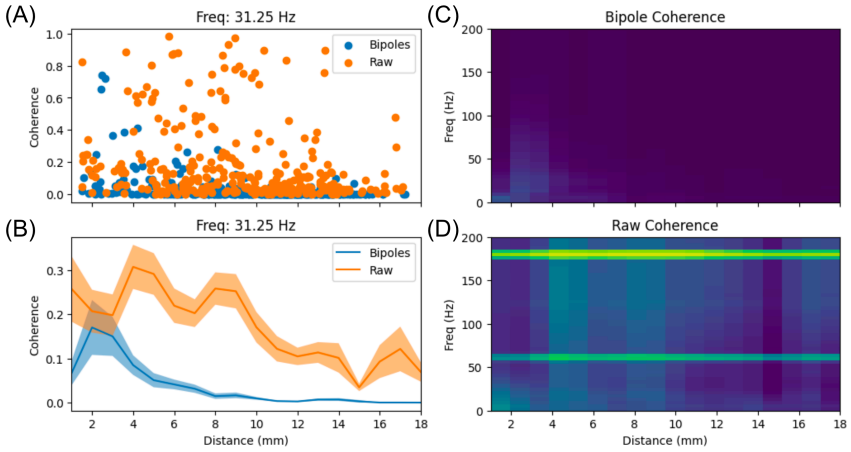

Figure 2.5: Coherence for 300 randomly selected pairs of electrodes and bipoles in the anterior drive of Monkey S on 2018-08-02. (A) Calculated coherence values for each pair at 31 Hz. (B) Binned average coherence plus or minus the standard error of the mean at 31 Hz. In (A,B), the coherence is much higher for the raw electrodes vs the bipole subtracted data. For all frequencies, the coherence is smaller for the bipole subtracted data (C) compared to the raw electrode data (D). The color scale is from 0 to 1. The bipole subtracted data also notch filters 60 Hz harmonics.

2.5 Time-Frequency Analysis

Due to the innate spectral properties of the LFP, it's desirable to perform time-frequency analysis. At any particular time in the task, one would want to know the power, and possibly phase, at all frequency bands. Unfortunately, the uncertainty principle tells us that it is impossible to simultaneously have high accuracy in both the frequency and temporal regime. If there is an increased accuracy in the timing, the frequency content is blurred. If there is increased accuracy in frequency, the temporal content is blurred.

There are many spectral analysis methods. Multitaper methods introduce an orthogonal set of tapers, typically Slepian sequences, for estimating power spectra [129] [130]. Empirical Mode Decomposition [131] [132] has been introduced to decompose the data into intrinsic mode functions that have nice properties for computing exact frequencies. Wavelets of different sizes can be geometrically combined to form superlets [133], which show an increased resolution in both time and frequency. I focus on two standard methods in the neuroscience community. The majority of analysis I perform use the Hilbert transform. However, for spectrograms, I typically use the Morlet Wavelet Transform.

Hilbert Transform Method

Formally, the Hilbert transform [134] of a signal $u(t)$ can be thought of as a convolution with the term $\frac{1}{\pi t}$:

$$H(u)(t) = \frac{1}{\pi} \lim_{\epsilon \to 0^+} \left[\int_{-\infty}^{t-\epsilon} \frac{u(\tau)}{t-\tau} d\tau + \int_{t+\epsilon}^{\infty} \frac{u(\tau)}{t-\tau} d\tau \right], \tag{2.3}$$

where the limit is needed to avoid the singularity when $t = \tau$. This has the nice property of transforming a signal with frequency ω 90 degrees. That is, the Hilbert transform of $e^{i\omega t}$ is

$$H(e^{i\omega t})(t) = e^{i(\omega t - \text{sgn}(\omega)\frac{\pi}{2})}, \tag{2.4}$$

where sgn takes the sign of its input. This gives a systematic way to transform signals, since any signal could, defined over a region, be expanded in a basis of exponentials.

The goal, however, is not the Hilbert transform of the signal. Instead, the goal is to find the power and phase of a signal. This can be achieved with the analytic signal. The analytic form $u_a(t)$ of the signal $u(t)$ is given by:

$$u_a(t) = u(t) + iH(u)(t). \tag{2.5}$$

The power at any time is given by the absolute value of $u_a(t)$ squared, and the phase is given by the angle of $u_a(t)$, or $\arctan(\text{Im}(u(t))/\,\text{Re}(u(t)))$.

An example is helpful. Let's say there is a signal given by $u(t) = a(t)cos(\omega t)$, where $\omega > 0$. The amplitude $a(t)$ of the signal is variable, but slow compared to the frequency of the signal ω. One could rewrite $u(t)$ in a basis of exponentials as

$$u(t) = \sum_n a_n e^{i\omega_n t}\cos(\omega t), \tag{2.6}$$

where $\omega_n < \omega$. By expanding, computing the Hilbert transform for each term, and collapsing, one finds the Hilbert transform as:

$$H(u)(t) = a(t)\sin(\omega t). \tag{2.7}$$

As expected, the power of $u_a(t) = u(t) + iH(u)(t)$ is then $a^2(t)$, and the angle of the signal is ωt. Thus, the Hilbert transform is useful for finding the power and phase of a signal.

Since the LFP has power in all frequencies up to 200 Hz, one first needs to bandpass the data. While computable, the power and phase is ambiguous when every frequency is present.

Morlet Wavelet Transform Method

The key point of this wavelet transformation is to convolve a Morlet wavelet with the signal of interest $u(t)$ to gather desired information. The Morlet wavelet is typically defined [135] as the product between a complex exponential and a Gaussian:

$$\Psi(f,t) = Ae^{i2\pi ft}e^{\frac{-t^2}{2\sigma^2}}, \tag{2.8}$$

where σ controls the width of the Gaussian and A normalizes the total energy to 1. By convolving $u(t)$ with the Morlet wavelet, one finds properties of $u(t)$ at any time t centered at any frequency f.

For example, let's say that $u(t) = a(t)\cos(\omega t)$, where $a(t)$ is slow compared to ω. One could expand this in terms of exponentials $a(t) = \sum_n a_n e^{i\omega_n t}$, where $\omega_n << \omega$. The Morlet Wavelet Transform $\tilde{u}(f,t)$ is:

$$\tilde{u}(f,t) = A\int_{-\infty}^{\infty} \sum_n a_n e^{i\omega_n t}\cos(\omega t)e^{i2\pi f(\tau - t)}e^{\frac{-(\tau-t)^2}{2\sigma^2}}dt. \tag{2.9}$$

With some tedious math, one finds this to be equal to

$$A\frac{\sqrt{2\pi}\sigma}{2}\sum_n a_n e^{i\omega_n \tau}\left[e^{-\frac{1}{2}\sigma^2(\omega_n+\omega-2\pi f)^2+i\omega\tau} + e^{-\frac{1}{2}\sigma^2(\omega_n-\omega-2\pi f)^2-i\omega\tau}\right]. \tag{2.10}$$

As mentioned before, $\omega_n << \omega$. Also, one is usually interested in regimes that are close to $2\pi f = \omega$, which means that one of the exponentials is much smaller than the other. This now simplifies to

$$\tilde{u}(f,t) = Aa(\tau)\frac{\sqrt{2\pi}\sigma}{2}e^{-\frac{1}{2}\sigma^2(\omega-2\pi f)^2}e^{i\omega\tau}. \tag{2.11}$$

This new analytic signal decays as the frequency f moves away from the true frequency $\omega/(2\pi)$. Also, one can examine the amplitude and phase.

The difficulty here is the trade off between frequency and temporal resolution. Larger σ's lead to a better frequency resolution but worse temporal precision. Typically, precision is often referred to in terms of the number of cycles n present in the Morlet wavelet [135]:

$$n = 2\pi f\sigma. \tag{2.12}$$

This method keeps the number of cycles constant, usually around 7.

2.6 Information Flow

A common method used in this thesis is averaging phase vectors to indicate information flow. For some time-locked point for each trial, one can take the Hilbert transform in a frequency band. The phase of the Hilbert transform can be useful in explaining which electrodes, and therefore brain areas, lead which other electrodes and brain areas. The thought experiment is as follows: Begin with an LFP signal originating in area X with some frequency. If this signal propagates through the medium, the phase will be approximately the same in area Y. If the signal passes through neurons, which takes significantly longer, then the "cycle" begins later and phase differs between areas X and Y.

One sample is not enough to indicate significance. I apply inter-trial phase clustering process [6] to

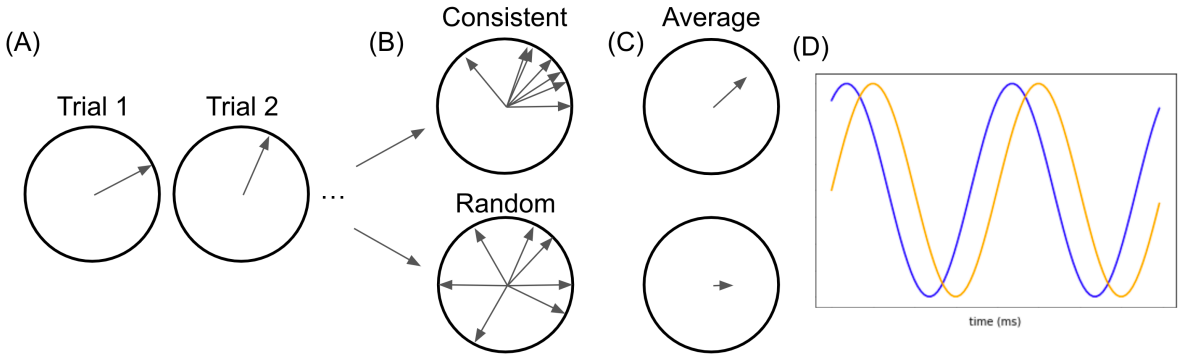

Figure 2.6: Example schematic of averaging the phase vector. (A) Individual phases per trial, which can either be the phase of an electrode or a phase difference between two vectors. (B) Overlaying multiple phases together. If the phase vectors roughly point in the same direction (top), they have a consistent phase. This leads to a large amplitude average phase vector (C, top). If pointing in random directions (bottom), the average amplitude is small (C, bottom). (C) The average phase vector. The amplitude relates to significance. The phase of this vector relates to the phase difference between the two channels. The phase difference may be random when the amplitude is small. (D) Example relation of two signals if there is a consistent phase difference: the blue trace "leads" the yellow trace.

phase differences. The idea is as follows: First, subtract the phases between two electrodes for every single trial. This resulting phase can be plotted on the unit circle (Fig. 2.6(A)). If the phase vectors consistently point in the same direction (Fig. 2.6(B, top)), the signal from one electrode consistently leads the signal from the other electrode. If they point in random directions (Fig. 2.6(B, bottom)), then the signals are unrelated between the two electrodes. All of the phases θ_j are averaged together with:

$$\overline{\theta} = \frac{1}{N}\sum_{j}^{N} e^{i\theta_j}, \tag{2.13}$$

where there are N trials. If the amplitude $|\overline{\theta}|$ is large enough, then the signal is significant (Fig. 2.6(C, top)). Otherwise, the amplitude is small and there is no significance (Fig. 2.6(C, bottom)). If significant, one can look at the phase of the average vector $\angle\overline{\theta}$ to see the amount that one electrode leads the other (Fig. 2.6(D)).

Calculating significance in terms of the amplitude $|\overline{\theta}|$ has been solved by Kluyver [136, 137] in response to a question posed by Pearson [138]. The probability or p-value that the amplitude $|\overline{\theta}|$ is larger than some fractional threshold r is given by:

$$P(|\overline{\theta}| > r|r, N) = 1 - rN\int_0^\infty (J_0(x))^N J_1(rNx)dx, \tag{2.14}$$

where J_i is a Bessel function of the first kind. The significance increases rapidly when N increases. For example, for about 700 trials, a threshold of $r = 0.125$ corresponds to a p-value of $\sim 10^{-5}$.

Periodicity is a concern for finding information flow. If the propagation time is about one period, it's ambiguous as to which signal leads the other. Ideally, the propagation time is at minimum less than

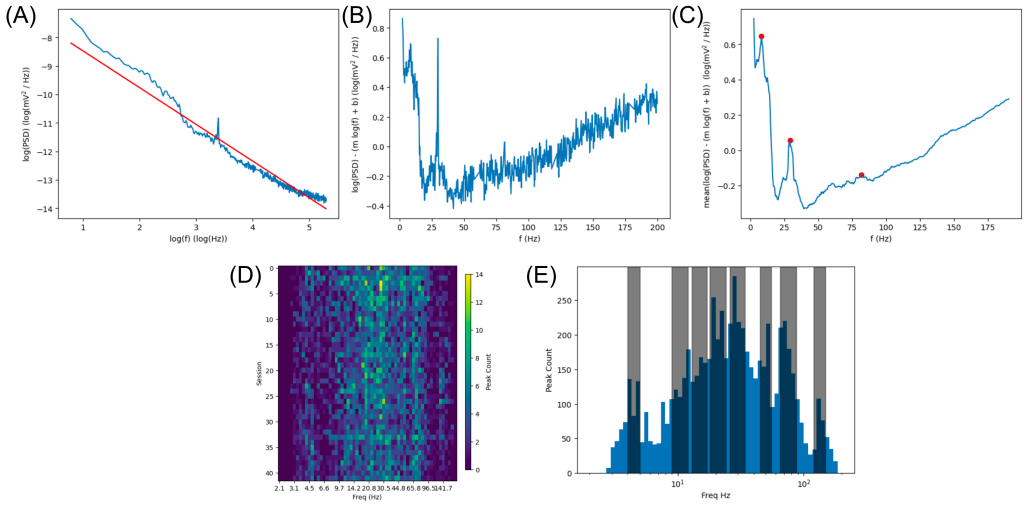

Figure 2.7: Method for calculating the frequency bands of interest. Panels (A) through (C) were calculated for the bipole from example channels in Monkey S's recording from 2018-08-17. (A) The power spectrum density along with the linearly regressed fit in log space. (B) The decay subtracted signal. (C) The averaged decay subtracted signal along with the peaks calculated from the peak fitting algorithm. (D) The histogram of peaks for all sessions or all bipoles across frequencies. (E) The grand histogram across sessions along with the selected frequency bands.

half a period. There are two ways to accomplish this. First, examining two electrodes that are close together will minimize the propagation time. Second, lower frequencies have higher periods, and more propagation time is needed before there are concerns.

What then, are the feasible limits of frequency ranges of concern? We can approximate the maximum possible delay from neural conduction velocities, estimated in [81] to be of order 18 m/s. Approximating the maximal distance between brain areas to be the size of the macaque brain or around $55mm$ [139], the estimated maximal delay is about 3 ms. Since this delay should be less than half a period, this corresponds to a frequency of 170 Hz. This limit is well above the frequency bands that are defined here, so this metric for information flow shouldn't run into periodicity concerns. That being said, one can be less concerned with smaller frequencies.

2.7 Frequency Band Selection

As discussed briefly above, different literature sources define different frequency bands. A data-driven approach is desirable, since different species and subjects may have slightly different natural frequency ranges. One approach [1] can be formed based on peaks in the LFP spectral content. Even though the power decays in a power law fashion, there are areas in the frequency spectrum that are larger than this baseline. By examining the power law for every electrode, one can find where there are peaks in the

[1] Thanks to Steven Hoffman and Charlie Gray for introduction to this method

power. Then, one can determine sensible frequency bands to analyze.

After the initial bipole subtraction, normalization, and notch filtering of the 60 Hz harmonics (sec. 2.8), the data from all bipoles and all sessions of interest is segmented into a 5 minute time region starting on the first trial. These 5 minutes occur when the subject is solely performing the task, and it includes cross fixations, feedback, object selection, etc. The power spectral density (PSD) is plotted with the Welch algorithm with the "nperseg" equal to 4096. The PSD is then limited to less than 200 Hz and outside a ± 2 Hz range from the 60 Hz harmonics. An example PSD is shown in Fig. 2.7(A). The $1/f$ power law is calculated with a least squares linear regression on the log-log PSD. That is, the PSD is transformed by taking the natural log of both the frequency and power. This line is also shown in Fig. 2.7(A). The power law decay is subtracted from the PSD (Fig. 2.7(B)). Then, a moving average is applied with a variable window size. A larger window is needed for large frequencies such that the averaged PSD has an equal sampling the log space. I set the windowing size to increase linearly with frequency: the smallest window size is set to 5 points and the largest window size is set to 80 points. This is done on both the x axis and y axis, where the $1/f$ decay has been subtracted via

$$y_{new} = \log(y) - (m \log(x) + b), \tag{2.15}$$

where m and b are the slope and intercept of the power law decay. The resulting average is shown in Fig. 2.7(C). From here, SCIPY's peak finding algorithm is used with a prominence of 0.03. The peaks of the PSD are shown in Fig. 2.7(C).

The peaks are calculated for bipoles in all sessions. In some cases, the power spectrum is extremely similar to a pure power decay. For this reason, I find bipoles where the difference between the maximum and minimum values of the decay subtracted PSD is less than 1; the peaks are dropped from these bipoles. I plot a histogram of the peak frequencies as function of session in Fig. 2.7(D). The peaks tend to be stable across frequencies, and the grand histogram is shown in Fig. 2.7(E). From here, one can, by eye, examine where frequency bands should be chosen. The vertical lines in Fig. 2.7(D, E) indicate where I think the frequency bands should be defined: 4-5, 9-12, 13-17, 18-24, 26-34, 45-55, 65-87, and 120-148 Hz.

One may be concerned that there are better methods for finding the PSD, especially given that different resolutions should be chosen for larger frequencies. However, I highlight that this method is solely used for frequency band choice. There may be a large amount of noise in peak location. Due to the large number of peaks, sessions, and bipoles included, a pattern emerges and sensible choices can be used. The bands are similar, but not exactly the same as in the literature (Sec. 2.5).

Another concern is that the chosen bands exclude some frequency ranges. It's entirely possible that some information is excluded. However, the chosen frequency ranges capture the most amount of

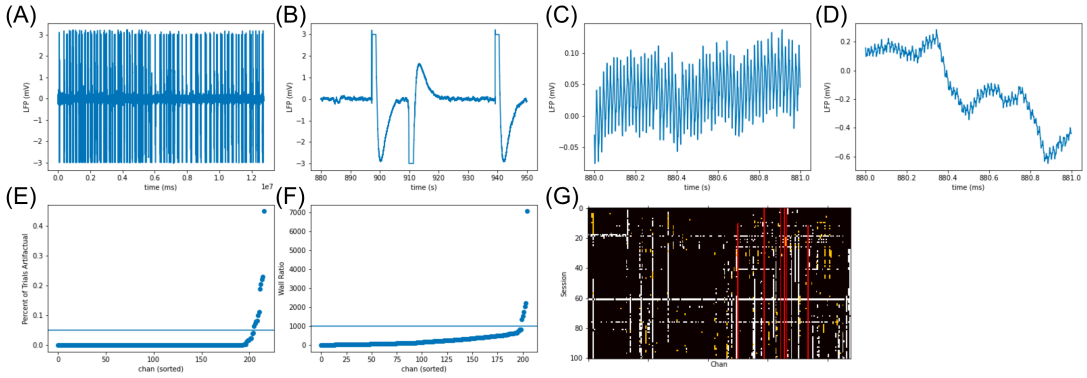

Figure 2.8: Examples of artifactual channels and methods for dropping channels. (A) The full LFP trace for example channel in session 2018-08-06, where the LFP often reaches the 3 mV recording limit. (B) Example artifact from (A). A relatively large (C) and normal amount (D) of 60 Hz noise present in example channels. (E) Percentage of trials with artifacts in session 2018-08-06 for each channel, sorted by percentage. The blue line indicates the threshold chosen for dropping channels; the threshold was chosen to drop a few dramatically artifactual channels. (F) Ratio of power from 60 noise to background power by channel in session 2018-08-06, sorted by the ratio. The blue line indicates the threshold to drop channels based on noise, chosen to occur at the elbow. (G) Classification of all channels in all sessions. Black squares indicate non-dropped data. Red indicates when the electrode was never lowered. Orange indicates channels that have too many artifacts. White indicates channels dropped due to 60 Hz noise.

information in the data.

2.8 Pre-processing Pipeline

This section describes the pre-processing pipeline of the neural data. The LFP is recorded at 30 kHz in the Buffalo lab. They lowpass filter the data at 500 Hz with a butter filter and then down-sample the data to 1000 Hz.

The data is then checked for artifacts. Some sessions are dropped based on bugged behavioral data or eye data. Channels that are not lowered into the brain/have unknown locations are dropped. Then, channels must meet two benchmarks. Occasionally, the LFP reaches the limit at about 3 mV (Fig. 2.8(A,B)). The LFP is considered artifactual even before it reaches the limit. This is a huge problem for frequency based analysis since oscillations cannot occur during flat regions. Further, there is a huge spike in power at all frequencies when the LFP transitions to and from a flat line. For every channel and trial, I flag when the LFP increases above $\pm$ 1.5 mV. Trials are defined to start 500 ms before the cross turns on and ends 500 ms after feedback. I.e. 500 ms after all reward has been given or 500 ms after the timeout period in incorrect trials. The extra 500 ms before and after the trial compensate for the artifactual behavior before and after the LFP increases past $\pm$ 1.5 mV. Channels are dropped if they have more than 5% artifactual trials. The 5% threshold is chosen due to an example session's channel artifact percentage: see Fig. 2.8(E) for details. The trials are flagged if they have at least one remaining channel with an artifact present.

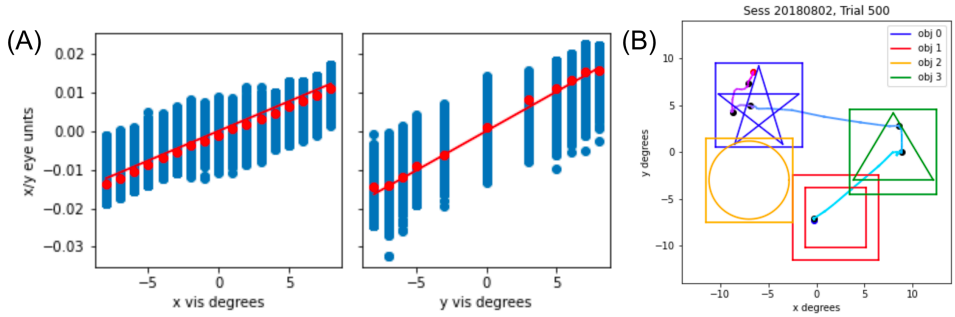

Figure 2.9: Calibration of scaling. (A) For each bounding box labeled in visual degrees, the eye position (blue) and average eye position (red points). The lines show the fit line. The slope of this line renormalizes the eye data to units of visual degrees. (B) Example eye trace when the objects and object bounding boxes are on the screen. The trace flows from blue to purple. Saccades, or eye movements, begin at the black dots.

Second, channels are dropped if there is too much 60 Hz noise [2]. Channels may be broken, which leads to large amounts of 60 Hz wall noise (Fig. 2.8(C,D)). To find the level of noise, the background power at 60 Hz is found by averaging the power from 59 and 61 Hz. The power at 60 Hz is calculated. The ratio of power to background power at 60 Hz is calculated, indicating the amount of noise injected into the system. Channels are dropped if the ratio is greater than 1000. This threshold was chosen based on all channels ratios; see Fig. 2.8(F) for details.

The plot of dropped channels/sessions is shown in Fig. 2.8(G). In some cases, sessions have a large percentage of their channels dropped. If a session has more than 73 dropped channels, the session is dropped. This removes 3 of monkey S's sessions: 2018-08-29, 2018-11-20, and 2018-12-14.

Bipoles are formed by finding nearest neighbor pairs (Sec. 2.4) and subtracting the signal. They are then normalized by z-scoring across the entire session. Then, notch filters are applied every 60 Hz up to 180 Hz.

Next, the power spectra densities of the bipoles are computed for all sessions. This leads to a definition of the frequency bands of interest (Sec. 2.7). For each frequency band, the bipole data is bandpassed with a high-pass and low-pass butter filter of order 5. For each bandpassed data, a Hilbert transform is applied to find the power and angle of each bipole (Sec. 2.5).

2.9 Eye Movement Processing

While the eye data was simultaneously recorded with the neural and behavioral data, the calibration was not recorded. The object placements and bounding box size wer recorded. The bounding boxes span 9 visual degrees and are centered at the object placements.

The center of the screen is determined via the cross fixation, where the eye location is within the

[2] Thanks to Jon Rueckemann for discussion regarding wall noise artifacts.

cross bounding box for 350 ms. For each session, the average eye position is calculated across all 350 ms for all trials. Subtracting the mean centers the eye data. For the final object fixation, the eye must be within the objects bounding box. Since the bounding box position is known, a line can be fit across all eye positions within the 800 ms fixation time for both x and y directions. The line is fit with no intercept due to the earlier mean subtraction. The eye data is re-normalized by dividing by the slope of the line (Fig.2.9(A)).

Next, the location of the eye compared to each stimulus is desired. While the bounding boxes are recorded in visual degrees, the exact stimulus position is not recorded due to the odd shapes that occur. Further, an image is portrayed to the screen that is approximately 200/24 times the size of the bounding boxes in both cardinal directions [3]. This scaling is needed for finding the exact stimulus locations on the screen in comparison to eye locations.

By eye and by defining r as the distance from the center to every edge of the image, I estimated the size of each shape. I estimated that the circle's four cardinal directions touch the edge of the image. The edges of the square occur at $0.76r$ in all four directions. The triangle touches the top center of the image, the bottom edge occurs at $0.72r$ below the center, and the bottom corners touch the left and right sides of the image. The star's top point touches the top edge of the image in the center. The far left and right points touch the sides of the image at $0.28r$ above the center. The bottom left and right points touch the bottom of the image at about $0.6r$ to the left and right of the center. With these values, approximate stimulus locations can be calculated in relation to the eye movement trace (Fig. 2.9(B)). The bounding boxes and stimuli CAN overlap (e.g. Fig. 2.9(B)). For this reason, it may be more significant to look at when the eye is within the stimulus instead of just the bounding box.

It's useful to say when the eye is within the bounding box and when it's within the stimulus. A position is within a circle if within the radius. One can determine if a position is within a square by checking the x and y positions relative to the edges. The triangle and star are more difficult, but can be solved by defining the lines connecting each point/corner and checking if a point is to the left of the line. For a line going from (A_x, A_y) to (B_x, B_y), a point (C_x, C_y) is left of the line if the cross product $\overline{AB} \times \overline{AC} > 0$ or $((B_x - A_x) * (C_y - A_y) - (B_y - A_y) * (C_x - A_x)) > 0$. One then needs to check the logical statements for all lines to see if a point is within a triangle or star.

Lastly, one needs to know when the eye moved. A saccade, or eye movement, is quantified by the distance traveled divided by length of time. To find this, I first smoothed the x and y eye traces by averaging consecutive 10 ms eye trace values together. This leads to an average eye speed, calculated from the distance traveled every point. Anytime the eye velocity is greater than 0.2 times the standard deviation of the eye velocity, I counted this as a significantly large eye movement. Combining these times into consecutive parts of eye movement define the complete saccade. E.g. one may see the sequence

[3] Thanks to Mike Jutras for measuring this scaling value.

$0, 1, 1, 1, 0$ where 1 indicates movement and 0 indicates no movement, the middle three points define the saccade. As long as there is less than 5 ms of time of no movement between two patches of movement, the entire thing is counted as a saccade. This is done to reduce noise. The saccade times can be used to define the start, end, direction, and distance of each times. If desired, one can look at the maximal distance traveled for the peak movement of the saccade too. The saccades are shown in the example trace in Fig. 2.9B. Note that the subject may saccade to the same object multiple times.

Stereotypical Saccades

After the subject finishes fixating on the fixation cross and before the objects turn on the screen, the subject usually saccades downwards in a stereotypical fashion. This is problematic since future analyses may deem this stereotypical movement as a saccade to a stimulus. These saccades are characterized to avoid future inclusion. Stereotypical saccades are defined by their direction, distance, and timing in the task. They may also be part of a multiple rapid saccade sequence, where the eye pauses for a very short amount of time before resuming motion.

To dissociate between saccades that are influenced by visual information and ones that are stereotypical, the timing of consecutive saccades is needed. In Sec. 3.3.1, saccades occur about once every 200 ms. Faster saccades occur in about 100 ms. With this in mind, if the intersaccade time is less than 100 ms, it is most likely part of one long saccade. Also, it should take at least 100 ms after the visual stimuli turn on before the subject can process the information. It takes roughly 100-130 ms for information to reach the PFC for judgement and decision making [140].

Stereotypical saccades are then defined as the following. First, the initial saccade must begin after the end of the cross fixation and before 50 ms after the objects appear on the screen. 50 ms is chosen to be sufficiently earlier than a 100 ms minimum visual processing estimate. Then, consecutive saccades with less than a 50 ms intersaccade time is combined with the initial saccade to form the ”stereotypical saccade.” Lastly, the direction of each individual saccade must be downward, defined as between -70 and -110 degrees. If, for example, the first saccade of two is downward and the second is not, only the first saccade is counted as stereotypical.

Chapter 3

Introductory Data Analysis

This chapter focuses on introductory data analysis to broadly task behavior and basic neural data. Starting with optimal task behavior, one can ask what expected results will occur when the subject performs optimally. We then compute a plethora of basic behavioral statistics, such as learning speed and errors during criterion trials. The visual search strategy is then examined, where unique strategies occur after specific conditions. Finally, introductory neural analysis leads to a basic understanding of the neural data.

3.1 Optimal Task Behavior

One topic of interest is understanding ideal task performance when memory limited or choosing features with or without integrating all stimulus information. As will be shown in chapter 4, subjects are memory limited. Is it necessary for subjects to remember more information? And is it necessary to integrate all stimulus information, the three features on each card, into a choice? For example, in non-human species, the motivation in solving the task is to receive an adequate amount of reward per effort spent. At some point, expending more energy to remember more information may not efficiently return more reward. This is complicated further by other correlates, such as arousal, that can lead to better or worse performance. This topic is also relevant for future task implementations, where one may encourage more or less engagement by tuning (e.g.) timeout penalties or reward amounts.

The discrete memory model, introduced in Sec. 4.3.2, is useful in understanding aspects of ideal behavior. It naturally allows for any memory limit. Noise is easily manipulated to observe when the subject is more or less focused on the task. By examining the total obtained reward and average block length in simulations, in some cases it's desirable to have less objects in memory, choose features without integrating information from seeing the cards, and focus less without loss of reward.

The discrete memory model is defined with slight adjustments to Sec. 4.3.2 such that statistics from

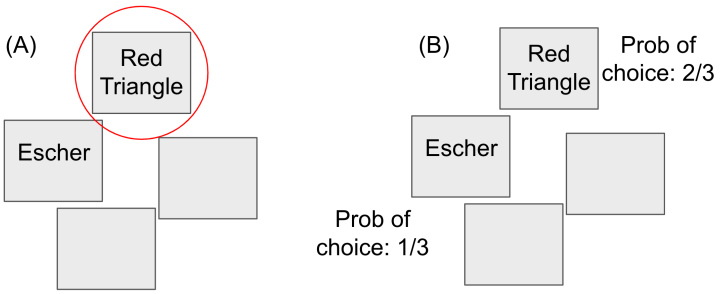

Figure 3.1: Schematic of choosing AFTER (A) or BEFORE (B) seeing the cards. In this example, the features red, triangle, and escher were all previously rewarded. If a subject chooses AFTER (A), they would choose the red triangle card 100 percent of the time. If the subject made a choice before seeing the cards (B), there would be an equal chance of choosing any of the three features; this leads to a 2/3 chance of choosing the red triangle card.

ideal behavior are easier to interpret. This model uses a belief vector $\mathbf{v} \in \mathcal{Z}^{12}$, filled with zeros, up to N_C ones, and up to N_I negative ones. The vector is initialized to zero. Features with $v_i = 1$ are preferentially chosen while features with $v_i = -1$ are preferentially avoided. Features with $v_i = 0$ are considered to have unknown information. N_C and N_I add memory limitations. These values highlight the memory limited ideal behavior of the subject: the agent can perform perfectly based on the features that are remembered.

To make a choice, the agent either chooses features "before" or "after" seeing the cards. By choosing before, the agent will choose a feature that maximizes the chance of getting rewarded. By choosing after, the agent will integrate information from the cards, where it will choose the card that maximizes the chance of getting rewarded. A schematic is shown in Fig. 3.1. Further, noise is added to each choice to mimic a lack of focus. If the agent is "not focused," it may choose randomly. Altogether, the probability P of choosing a card k on a trial t from a set of one hot encoded cards $\mathbf{C} \in \mathcal{Z}^{4\times 12}$ becomes:

$$P_k^{(t)} = \begin{cases} (\epsilon)\frac{1}{3} + (1 - \frac{4}{3}\epsilon)\frac{\mathbf{c}_{k;}^{(t)} \cdot \mathbf{v}^{(t)}}{\sum_{j=1}^{12} v_i^{(t)}} & \text{if } \sum_i v_i > 0 \\ (\epsilon)\frac{1}{3} + (1 - \frac{4}{3}\epsilon)\frac{3 + \mathbf{c}_{k;}^{(t)} \cdot \mathbf{v}^{(t)}}{12 + \sum_{j=1}^{12} v_i^{(t)}} & \text{if } \sum_i v_i < 0 \\ \frac{1}{4} & \text{if } \sum_i v_i = 0 \end{cases} \tag{3.1}$$

if choosing before seeing the cards and

$$P_k^{(t)} = (\epsilon)\frac{1}{3} + (1 - \frac{4}{3}\epsilon)\delta(k - \text{argmax}(\mathbf{C}^{(t)} \cdot \mathbf{v}^{(t)})) \tag{3.2}$$

if choosing after seeing the cards. If "argmax" returns multiple cards, a random card is selected. If there is a 100 percent chance of choosing a particular card without noise, than there is a 0 percent chance of

choosing that card with 100 percent noise.

Ideal logic is used to update the hidden state $\mathbf{v}$ on every trial. The following definitions are used: "rewarded features" ($v_i = 1$) previously gave a positive reward. Non-rewarded features ($v_i = -1$) previously gave a negative reward. Forgetting is setting v_i to 0. The logical reasoning is the following: if correct and the chosen card has 1-3 rewarded features, forget about any non-chosen feature. If correct and the chosen card has 0-2 non-rewarded features, remember chosen features that were not previously non-rewarded. If incorrect, choosing a card with rewarded features, and there exist non-chosen rewarded features, remember the non-chosen rewarded features. If incorrect, choosing a card with rewarded features, and there does not exist non-chosen rewarded features, forget everything and remember that the chosen features are all non-rewarded; this typically occurs when the rule changes. Lastly, if incorrect, 0-3 non-rewarded chosen features, and the non-chosen features are all also either unknown or non-rewarded, then add the chosen features to the list of non-rewarded features. If incorrect, 0-3 non-rewarded chosen features, and there exists non-chosen rewarded features, then do nothing to the remembered features. The memory limitation is added after the logic of computing $\mathbf{v}^{(t+1)}$: if the number of positive (negative) ones is greater than $N_C(N_I)$, the model randomly selects $N_C(N_I)$ positive (negative) ones and sets the rest to 0.

Memory decay is also added to correct features when $N_C > 1$, which is a natural way to simulate "forgetting" of features. A memory decay is added by setting the probability to remember each feature on every trial. A randomly selected remembered feature is retained. The probability of keeping any other feature in memory is set to p. This leads to forgetting of all but one feature after enough trials.

Insights from Simulations and their Rewards

Statistics are aggregated for 20 simulations of 1000 trials, using the same task parameters as the monkeys. A "reward" value of 1 is given on every correct trial. Incorrect trials receive a value of 0. The reward value can be used to determine the percentage of correct trials.

For the first set of simulations, the agent chooses after seeing the cards and there is no memory decay. The total reward received and average block size is shown in Fig. 3.2(a,f) for different amounts of memory. Increasing memory capacity leads to more reward and smaller block sizes. Although, adding memory capacity is more helpful when there is less memory capacity. E.g., one receives more reward by increasing from 2 to 3 incorrect objects in memory compared to the increase from 8 to 9 incorrect objects. The largest increases in reward occur by adding memory capacity to correct objects instead of incorrect objects. Thus, if a subject was trying to save energy, focusing on correct objects is more efficient. It's worth noting that the average block size barely decreases when $N_C = 3$ from 14 to 12, indicating only minor average improvements.

To compare rewards when choosing before or after seeing the cards, the agent uses a variable number

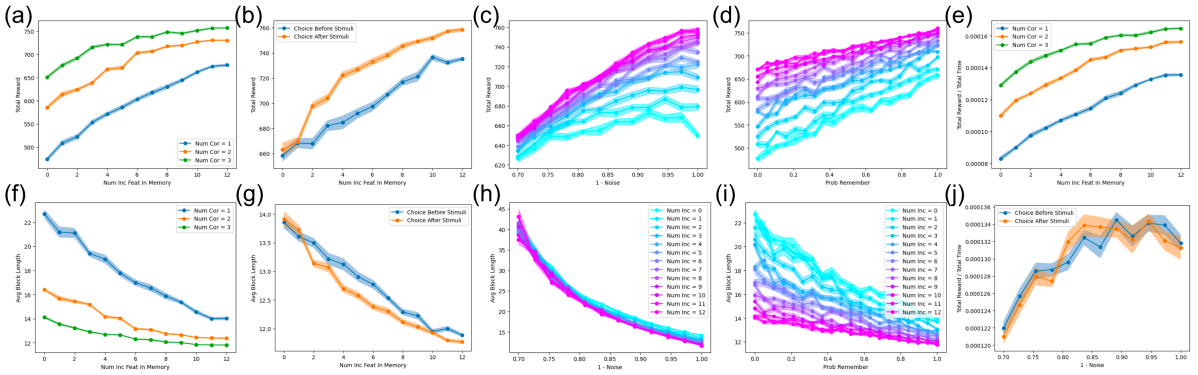

Figure 3.2: Rewards and block sizes for variations of the discrete memory model. (a,f) Total reward and average block size for perfect agents with varying amounts of correct (N_C) and incorrect (N_I) memory. (b,g) Total reward and average block size for $N_C = 3$ and varying N_I when the agent chooses before or after seeing the stimuli. There is no benefit to choosing after seeing the cards when $N_I \leq 1$. (c,h) Total reward and average block size for varying amounts of noise for $N_C = 3$. For small N_I, the agent gains more reward with a small amount of noise. (d,i) Total reward and average block size for a varying memory decay when $N_C = 3$. (e) shows plot (a) if reward per total time is used. (j) shows the $N_I = 0$ case for plot (c) if reward per total time is used; the reward ratio flatlines instead of peaking.

of incorrect features in memory and $N_C = 3$, which is chosen due to the efficient performance in Fig. 3.2(a). The total reward and average block size is shown in Fig. 3.2(b,g). For $N_I \leq 1$, there is no difference in performance when choosing before or after seeing the cards. However, when $N_I > 1$, it is strictly better to choose after seeing the cards. Thus, if $N_I \leq 1$, it's more efficient to choose before seeing the cards; it takes more effort to integrate the visual information with beliefs to make a choice. However, if $N_I > 1$, it is better to choose after integrating visual information.

Noise, mimicking a lack of focus, is now added to the agent for $N_C = 3$ and a variable N_I. The total reward and average block size is shown in Fig. 3.2(c,h). The average block size is drastically affected by the amount of noise while the total reward is affected less. When N_I is large, noise affects the total reward more. When N_I is small, it is surprisingly advantageous to have noise. This most likely is due to the agent reaching the 16 out 20 criterion instead of the 8 out of 8 criterion for a rule switch. With this threshold, one could hypothetically game the system by purposely failing the task to get more total trials correct. This has a profound affect on efficiency. If a subject was only holding correct features and 0–1 incorrect features in memory, they are not affected with a small lack of focus.

Memory decay, where there is a probability $1-p$ of forgetting a correct feature, is added to the agent when $N_C = 3$. The total reward and average block length is shown in Fig. 3.2(d,i). Remembering all correct features leads to strictly more reward. Although, decreasing p by a very small amount, such as 0.1, will not lead to a drastic decrease in total reward gained.

The question remains if there are differences based on the time-out period that the monkeys faced.

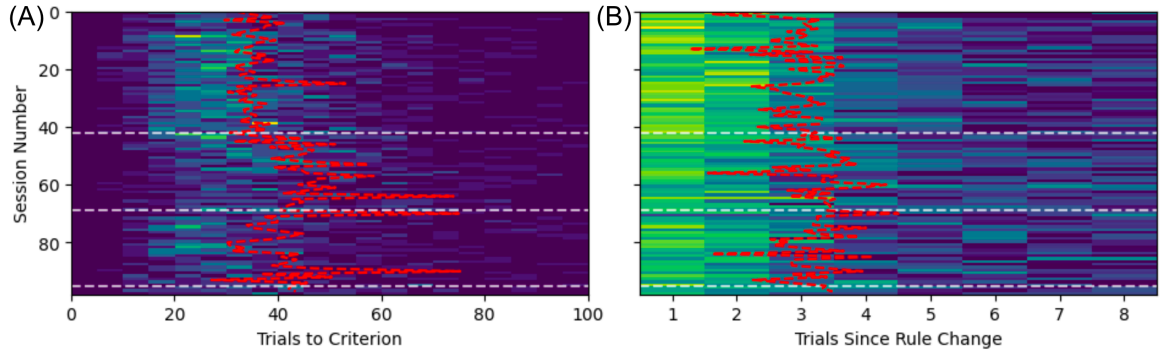

Figure 3.3: Behavioral statistics showing a lack of improvement in the task over all sessions. Session number is on the y-axis, where the white lines indicate task modification. Between sessions 42-69 and after session 95, the timeout period was shortened. Between sessions 69-95, different stimuli were presented. (A) Histogram of the trials to criterion with the average trials to criterion overlaid in red. The average is fairly consistent until the timeout period was shortened. Afterwards, the subject performs slower. (B) Percentage of times the subject chose a card with the previous rule feature as a function of the trials since the rule change. Chance (dark-blue) is 25%. The red line indicates the average per session.

One needs to calculate the reward per total time, which accounts for both reward and the penalty. The reward is still 1, which is equivalent to some arbitrary amount of chow received. The total time is the average length of a trial, calculated as 2200 ms, plus 1400 ms for correct trials and 5500 ms for incorrect trials.

Two figures are replotted with the reward per total time in Fig. 3.2(e,j). First, like in Fig. 3.2(a) the total reward per total time still increases with memory capacity. Second, the total reward per total time flatlines for the change in noise, compared to Fig. 3.2(c). For the case of $N_C = 3$ and $N_I = 0$, with this penalty, one would receive the same amount of total reward when the noise is less than about 0.15. While it is no longer beneficial to have noise, it is still non-detrimental. In the current case, the timeout period penalty doesn't appear to change expected reward too much. Modifying the timeout period further could lead to further changes in the subjects strategy.

3.2 Behavioral Traits

There are many behavioral statistics that explain the behavior of Monkey S. While some are included in chapter 4 for their relevance in the behavioral strategy, others can help explain nuanced ideas of how the subject behaves in the task.

3.2.1 Learning Speed by Session

One target of the U19 collaboration is to study the ability for the subjects to "learn to learn." The idea is that the subject will gradually perform better at the task over a long time scale. However, the subject was taught the task before any behavioral/neural recordings occurred, so the data likely does not capture

this long timescale learning. To check if the subject, monkey S, learns on a long timescale, I examine both the trials to criterion and the perseveration of the previous rule.

Better task performance leads to less trials to criterion. The trials to criterion for each session, in Fig. 3.3(A), show that the average performance does not improve over long timescales. In fact, after the task is modified and the timeout period is shortened, the subject performs worse. After the timeout period is lengthened and new stimuli are presented, the subject does not return to their original performance level.

Better task performance should also lead to the ability to perseverate less. Perseveration is defined as consistently choosing the previous rule feature, even when getting trials incorrect. For every trial after the rule change, I find the number of times that the subject chose a card with the previous rule feature without regards to correct or incorrect information. After enough trials, the percentage p of perseverations should approach a chance value of 25%. In Fig. 3.3(B) the percentage of perseverations is consistent across sessions, even when the task is modified. The average number of perseverations is approximately three, calculated as $E(n) = \sum_n P(n)n$ where $P(n)$ is normalized as $P(n) = (p(n) - 0.25)/(\sum_j (p(j) - 0.25))$.

Unfortunately, there is no improvement in performance in either the number of trials to criterion or perseveration across sessions. With this in mind, one cannot look for signatures of "learning to learn" within this dataset. One can still look at the learning of the rule during each block. One could also look at the behavioral change when the task structure changes. It's possible that this influenced the behavior of the subject. It is interesting that task performance decreases when the timeout period is shortened (Fig. 3.3(A)), which can be thought of as "learning to learn." It's possible that, given that the negative feedback decreases, the subject received more reward leading to a new kind of learning.

3.2.2 Errors During Criterion Trials

There's an interesting behavior during seemingly random error trials at the end of rule blocks. When the subject reaches the criteria of 16 out 20 correct trials in a row, there are 4 incorrect trials (labeled "inc 1," "inc 2," etc.). In these trials, the subject typically chooses features that were previously correct, but not the rule feature. Statistically, is the subject "distracted" by other features? If so, the subject may use a strategy where all features are influenced regardless of time spent in the rule block.

Given that the trial before the error trial was correct, did the error trial include a feature that was chosen on the previous trial $t - 1$? The cases are as follows: After a correct trial, the two "distraction features" could appear on the same card, different cards, on the rule feature card, or not on the rule feature card (Fig. 3.4(A)). I focus on the two cases where there is one card with distraction features (middle two rows in Fig. 3.4(A)). If a random choice occurs, the subject chooses a card with the "distraction features" 1 out of 3 times. If the subject chooses this card more than 33% of the time, they are likely distracted by the other features.

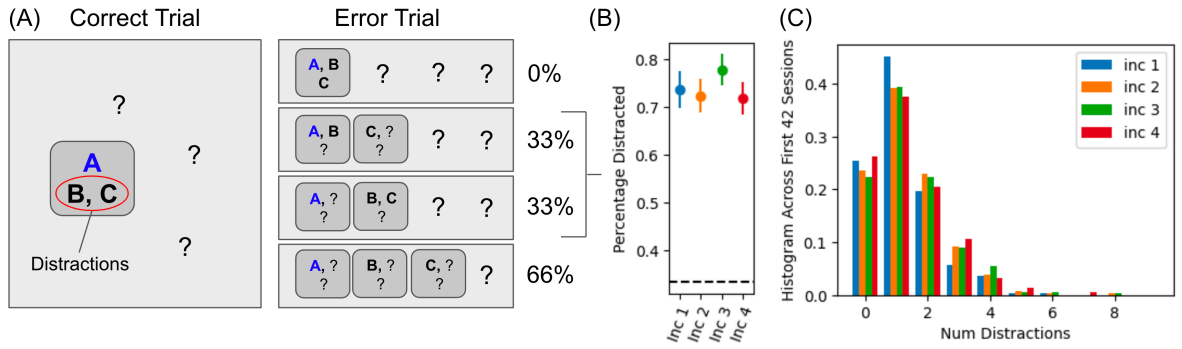

Figure 3.4: (A) Example schematic of distractions during the error trials. On a correct trial towards the end of the rule block, the subject chooses features A (the rule), B, and C. The subject should understand that the rule is feature A. During the error trials, the features could appear in any of the four cases shown. Given that the card with feature A is NOT chosen, the percentages show the likelihood of randomly selecting a card with features B or C. Selecting cases from the middle two panels, (B) shows the percent of the time that the subject chooses a card with a distraction feature, which is much higher than chance. (C) Number of times a feature is chosen before it is chosen during each error trial, leading to an average of 1.3 trials.

Each error trial is analyzed separately since there may be a difference depending on learning a particular rule. Only the 42 sessions before the shortened timeout period are used. In Fig. 3.4(B), the percentage of times ($\sim 75\%$) the distraction features are chosen is much higher than chance, convincingly showing that the subject is distracted. There are little to no differences between each error trial, indicating that this distraction mechanism can occur at any time while learning the rule.

Further, for every error trial, I look at the previous trials and find the feature that is chosen most often in a row. For every error trial, the histogram of number of times a feature is chosen before distracted (Fig. 3.4(C)) shows that, on average, the subject is distracted by choosing a feature 1.3 times. This indicates that recent information distracts the subject from choosing the rule feature.

This analysis shows an important aspect of the subjects' behavior. Given that they are readily distracted by receiving reward for non-rule features, they integrate all information regardless of time spent in the rule block. This opposes an ideal observer hypothesis, where the subject rejects information about non-rule features when they know the rule. This analysis supports the reinforcement learning strategy found in sec. 4.4.1, since the subject always integrates new information regardless of rule knowledge.

3.2.3 Behavioral States

To fully understand the subjects' strategy, one needs to ask specific questions related to their behavior. For example, what happens when they were previously correct? Incorrect? What if there are immediate trials that were incorrect? In this section, I ask a plethora of questions, one at a time, that help the understanding of the subjects' strategy. This leads to a depiction with a random state, hypothesis state, and exploit state.

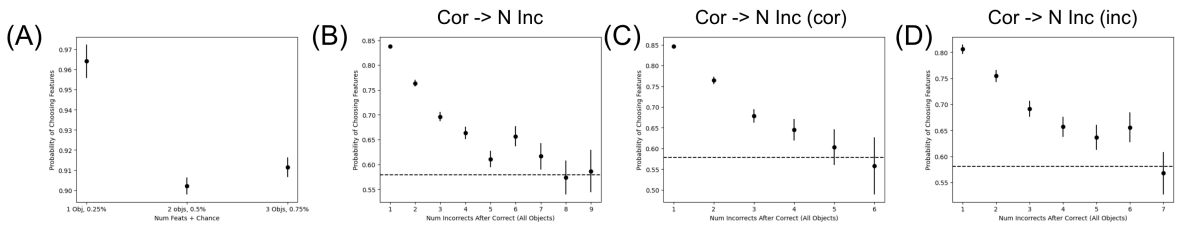

Figure 3.5: (A) Probability of choosing a card with previous correct features A, B, and/or C if they are split between 1, 2, or 3 cards. (B) Probability of choosing a card with previous correct features A, B, and/or C with N incorrect trials in between. (C) adds the constraint that, during every "in between" trial, the subject chooses a card with features A, B, and/or C. (D) is the same as (B) with the added constraint that, during the last "in between" trial, the subject does NOT choose a card with features A, B, and/or C. Dotted lines indicate chance.

For Monkey S, only the sessions before the shortened incorrect timeout period are used. For a specific condition or sequence of trials, all trials across all sessions meeting this condition are concatenated. Then, the mean and standard error of the mean are calculated based on the observed binomial distribution, where the subject could have chosen or not chosen a particular feature. Chance levels, or the chance that an observer who chooses randomly, are calculated via the observed conditions and trials and verified with theoretical calculations.

Preferring Recent Correct Features

How often does the subject choose a card with a previously correct feature? The following constraint is imposed: Trial $t-2$ was incorrect, trial $t-1$ was correct, and the subject chose features A, B, and C on trial $t-1$. On trial t, how often did the subject choose a card with features A, B, or C? This can be further divided on trial t: features A, B, and C could all be on 1 card, features A and B could be on 1 card while C is on a second card, or all three features could be on different cards. Chance is 25, 50 and 75% respectively. An example of feature splitting is shown in Fig. 4.5.

The likelihood of choosing the previously correct features is shown in Fig. 3.5(A). The subject chooses the previously correct features most often when they all appear on a single card. As argued in Sec. 4.4.2, this most likely occurs due to a preference towards the previous card if it was correct on trial $t-1$.

How long does this memory last? Specifically, if on trial t the subject is correct and chooses a card with features A, B, and C, does the subject choose a card with any of these features on trial $t+N+1$ given that trials $t+1, t+2, ..., t+N$ were incorrect? Again, the features on trial $t+N+1$ can be split between 1, 2, or 3 cards (Fig. 3.5(B)). After around 5 incorrect trials, the subject approaches chance levels. This also occurs if an incorrect trial is added to trial $t-1$. However, it is rare for there to be many incorrect trials after an incorrect and correct trial sequence.

To find the theoretical chance of features splitting between cards, one can calculate the probability independently for each feature. On trial $t+N+1$, features A, B, and C appear on a single card 1/16 times:

1. Find probability that A is on any card (100%)
2. Find probability that B is on the same card as A (25%)
3. Find probability that C is on the same card as A and B (25%)
4. Find probability of randomly choosing the card with A, B, and C (25%)
5. Calculate total probability from this case as $P_1 = 1/64$.

Features A, B, and C appear on two cards 9/16 times:

1. Find probability that A is on any card (100%)
2. Possibility 1: Find probability that B is on the same card as A (25%)
3. Find probability that C is NOT on the same card as A and B (75%)
4. Possibility 2: Find probability that B is Not on the same card as A (75%)
5. Find probability that C is on either card with feature A or B (50%)
6. Find probability of randomly choosing the card with A, B, and C (50%)
7. Calculate total probability as $P_2 = (1/4 \times 3/4 + 3/4 \times 2/4) \times 1/2 = 18/64$.

Features A, B, and C appear on three different cards 6/16 times:

1. Find probability that A is on any card (100%)
2. Find probability that B is NOT on the same card as A (75%)
3. Find probability that C is NOT on the same card as A OR B (50%)
4. Find probability of randomly choosing the card with A, B, and C (75%)
5. Calculate total probability from this case as $P_3 = 18/64$.

The total probability of choosing a card with features A, B, and/or C is $P_1 + P_2 + P_3 = 37/64 \approx 58\%$.

The question remains, WHY does the probability decay? It's possible that it decays due to the intrinsic memory of monkeys [96]. It's also possible, however, that they simply switch states deterministically and the average statistics look like a decay process. Imagine the following decision tree after a correct trial: p percent of the time choose a card with features A, B and/or C and $1 - p$ percent of the time choose randomly for all following trials. With this decision tree, the probability of choosing a card with features A, B and/or C would decay like $p_0 p^n$, where n is the number of trials after trial t_0.

One test is the following: redo the previous memory decay, but enforce either side of the branches in the above decision tree. First, if the subject chose a card with features A, B, and/or C on every trial

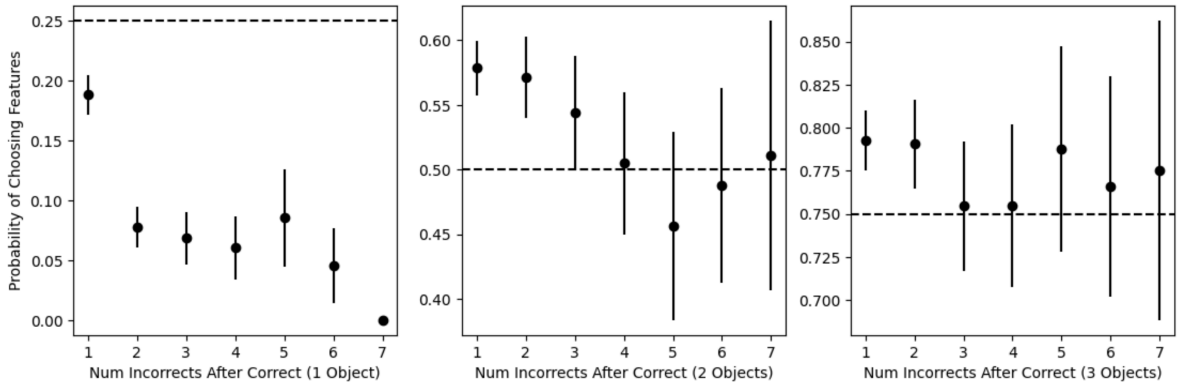

Figure 3.6: Probability of choosing a card with previously incorrect (trial $t+N$) features A, B, and/or C on trial $t+N+1$ when these features are split between 1 (left), 2 (middle), and 3 (right) cards. Dotted lines indicate chance. The subject only avoids these features when the card repeats (left).

until trial $t+N+1$, the probability on trial $t+N+1$ of choosing a card with features A, B, and/or C would remain constant. In Fig. 3.5(C), the probability decays. Second, if the subject did NOT choose a card with features A, B, and/or C on trial $t+N$, the subject should begin choosing randomly. In Fig. 3.5 (D), the probability decays. Thus, the decision tree hypothesis is ruled out.

From these analyses, Monkey S prefers to choose cards with previously correct features, and their memory decays over ∼5 incorrect trials. This aligns with the previously found timescales [96], where monkeys tend to forget all prior knowledge after an inter-trial interval of about 20-30 seconds. In this case, 5 incorrect trials, with an average of about 8 seconds, leads to a time delay of around 40 seconds. While slightly longer, the delay occurs at roughly the same time. There may be differences in task structure, such as repeatedly showing stimuli, that lead to differences in delay.

Avoiding Incorrect Card, NOT Features

Do subjects AVOID previously incorrect features? I examine a similar question to before, where the subject is correct on trial t and incorrect for trials $t+1, t+2, ..., t+N$. On trial $t+N$, the subject chooses features A, B, and C, which are incorrect. On trial $t+N+1$, how often does the subject choose one of these features? I check the cases between times where features A, B, and C are split between 1, 2, or 3 cards: chance corresponds to 25, 50, and 75%. If the subject avoids these features, they will choose cards with features A, B, and/or C less than chance.

In Fig. 3.6, the subject avoids these previous features only if they appear on a single card, or when the card repeats. Otherwise, the subject chooses the incorrect features at a higher rate than chance for 1-2 trials. As I'll show later, this is a perseverative effect, where the subject focuses on the previous CORRECT features for a few trials.

The subject tends to focus on correct features and the previously correct or incorrect card. This is a plausibly efficient solution to this task. Only correct features give reward, so they are more important

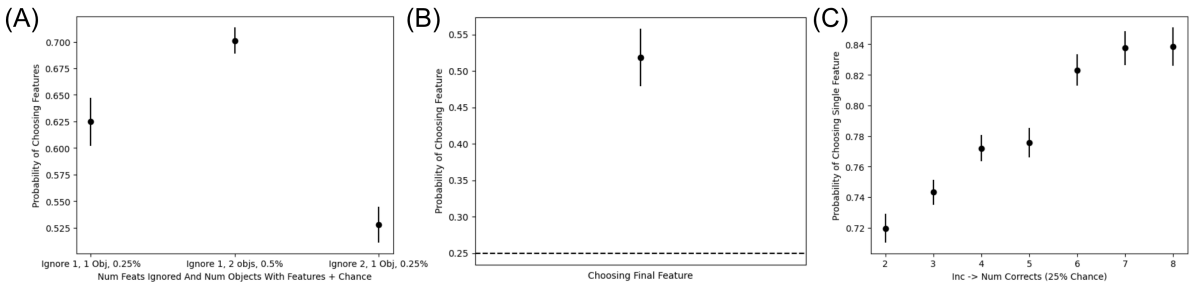

Figure 3.7: (A) Probability of choosing correct features A, B, and/or C after one or two features have been ruled out (ignore) and the remaining features are split between one or two cards/objects. (B) Probability of choosing last remaining feature after two other correct features have been ruled out over two trials. (C) Probability of choosing the single consistently correct feature after N trials.

to memorize. However, a previous card's feedback has the most reliable predictability of the next trials feedback. That is, if the previous card was correct or incorrect, there is an extremely high chance it will be correct/incorrect again. Thus, there should be at least two general behavioral states: a random state and a state focused on the previous correct features. At the same time, there is likely memory of the previous correct features.

Number of Features in Memory

How many correct features are held in memory? And is there a change of memory after an incorrect trial? I focus on the trial after an incorrect trial. The subject chooses incorrectly on trial $t-1$, features A, B, and C correctly on trial t, and then incorrectly on trial $t+1$. On trial $t+1$, at least 1 feature should be ruled out. How often does the subject choose the other features on trial $t+2$? There are three possibilities. The subject could rule out feature A on trial $t+1$, and features B and C appear on the same card on trial $t+2$. There is a 25 percent chance the subject randomly chooses this leftover card. Second, the subject could rule out feature A on trial $t+1$, and features B and C appear on different cards on trial $t+2$. There is a 50 percent chance the subject chooses features B or C. Lastly, the subject could have ruled out both features A and B on trial $t+1$. There is a 25 percent chance of randomly choosing feature C on trial $t+2$.

The probabilities for all 3 cases are shown in Fig. 3.7(A). The probability is higher than chance for all cases, and the relative probabilities are also as expected. If the average probability for choosing each feature is $P_A = P_B = P_C = P_x$, and the probability of choosing any of the nine other features is P_r, then the probability of choosing a card with features A, B, and/or C is $2P_x + P_r$ for the first case, $2P_x + 4P_r$ for the second case, and $p_x + 2P_r$ for the third case. This implies that the probability for the second case should be larger than the first case, which should be larger than the third case. These relative probabilities are found in Fig. 3.7(A).

Can the subject remember three correct features? The previous analysis is modified: the subject chooses incorrectly on trial $t-1$, features A, B, and C correctly on trial t, feature A incorrectly on trial

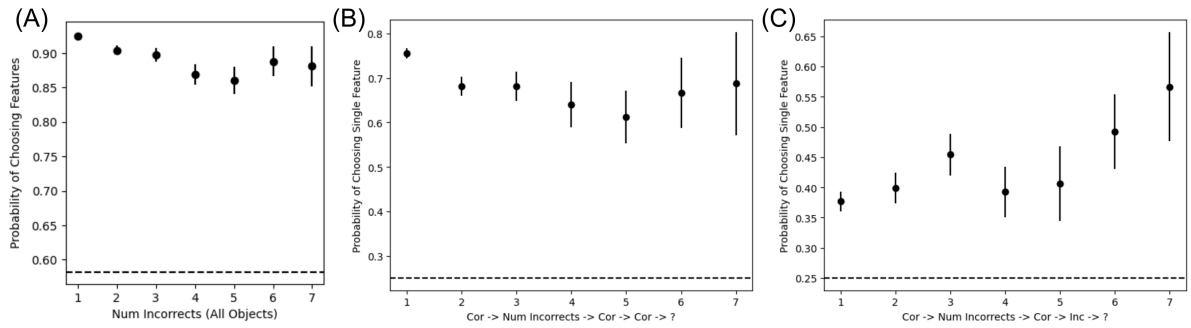

Figure 3.8: (A) Probability of choosing correct features A, B, and/or C after N incorrect trials. (B) Probability of choosing feature A when chosen correctly twice in a row after N incorrect trials. (C) Probability of choosing feature A after N incorrect trials; feature A was correctly chosen and then incorrectly chosen.

$t+1$, and feature B incorrectly on trial $t+2$. What then, is the probability of choosing feature C on trial $t+3$? Chance is 25 percent for a random observer. The probability is shown in Fig. 3.7(B), which is again much higher than chance and similar to the third case in Fig. 3.7(A). The subject is able to remember up to all three correct features.

Learning the Rule

Based on the previous analysis, the subject likely rules out incorrect features to slowly converge on the correct feature. How does the subject learn the correct feature? The analysis is as follows: on trial $t-1$ the subject is incorrect. On trial t, the subject chooses a card correctly. For trials $t+1, t+2, ..., t+N$, the subject chooses only feature A consistently. That is, feature A is the only feature that is ALWAYS chosen. Then, what is the chance that the subject chooses feature A again on trial $t+N+1$? A randomly selecting subject would choose feature A at 25 percent. The initial incorrect trial is used to initialize the sequence of correct trials.

In Fig. 3.7(C), the probability of choosing feature A increases slowly throughout the trials. One may expect that it converges to a constant eventually, which is not reached due to the 8 out of 8 trials to criterion. This also implies that the subject is constantly learning the rule, inconsistent with an ideal observer that reaches a plateau after only two correct trials.

These last few analyses lead us to the next piece of the puzzle. There are likely at least three behavioral states: a random state, a hypothesis state or one where multiple correct features are known, and a state of exploitation of a single feature.

Transitioning Out of Random State

The first transition to consider is one from the random state to the exploit state. While it's difficult to tell how many trials it takes to transition into the random state, one can examine the trend for N incorrect trials. The analysis is as follows: on trial $t-N-1$, the subject is correct. On trials

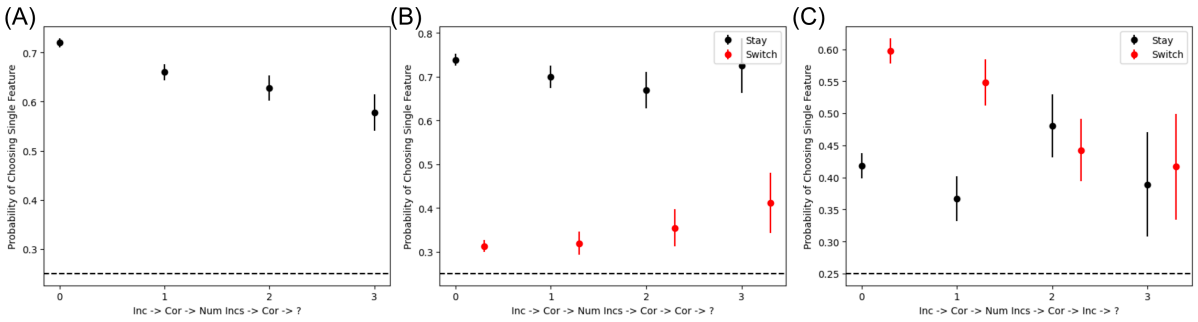

Figure 3.9: (A) Probability of selecting the consistently chosen correct feature when two correct trials are split by N incorrect trials. If there are two consistently chosen features, and one is correctly (B) or incorrectly (C) chosen again, (B,C) shows the probability of staying (choosing feature again) vs switching to the other feature.

$t-N, t-N+1, ..., t-1$, the subject is incorrect. On trial t, the subject correctly chooses features A, B, and C. What is the probability of choosing any of these features on the next trial? In Fig. 3.8(A), the probability remains relatively constant. The same trend is observed when the features are split between 1, 2, and 3 cards (not shown). This implies that the subject will quickly transition, no matter where they are in the random state transition process.

Next, I examine the case of multiple correct trials in a row, indicative of the exploit state. The same framework is used again, except that the subject chooses only feature A on trial $t+1$. What then is the probability of choosing feature A on trial $t+2$? As shown in Fig. 3.8(B), the probability is fairly constant. This shows us that the subject will quickly transition towards the exploit state if possible.

Finally, I look at exploitation after a single trial. One of the trademarks of exploitation is perseveration (Sec. 4.4.2). I check the first case of this section with the following constraint: on trial $t+2$ the subject incorrectly chooses only feature A. How often does the subject choose feature A again? An ideal agent should avoid this feature, while a perseverative agent would fixate on it. In Fig. 3.8(C), the subject chooses feature A above chance, indicating that they perseverate after a single correct trial.

Transitioning towards the exploitation state may occur due to the subject perseverating after a single correct trial. While it's possible that they immediately transition from the random state to the hypothesis state, they could also go towards the exploit state. With this in mind, I define a "minor exploit" state, where the "major exploit" state occurs after multiple correct trials in a row. Then, the subject transitions to the hypothesis state from the minor exploit state after a single incorrect trial.

Transitioning to the Exploit State

Is it possible to enter the exploit state if multiple incorrect trials occur between the multiple correct trials? I initialize the sequence with an incorrect trial on trial $t-1$ and a correct trial on trial t. The subject chooses features A, B, and C on trial t. Then, there are $0 \leq N$ incorrect trials. Finally, the subject is correct on trial $t+N+1$, where they chose feature A. What is the chance they choose feature A again? In Fig. 3.9(A), the probability is relatively constant from $N=0$ to $N=3$. Instead, the

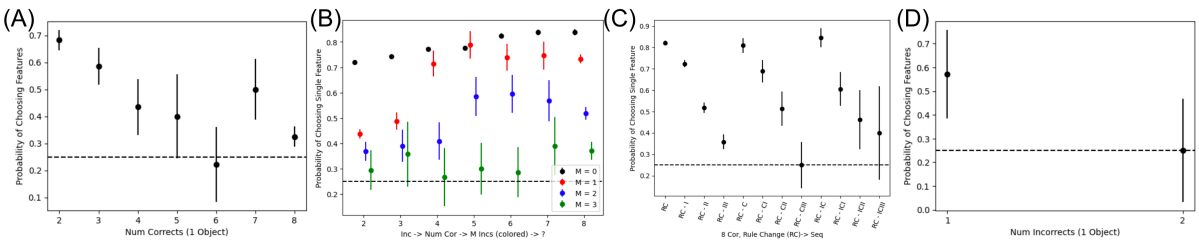

Figure 3.10: (A) Probability of choosing a feature from the most recent correct trial that was not the previous rule feature, given that the subject incorrectly chose the previous rule feature. In this case, the two non-rule features appear on the same card, but the same trend occurs when the two non-rule features appear on different cards. (B) Perseveration after Num correct trials and M incorrect trials. (C) Perseveration after specific sequences of trials: C refers to a correct trial and I refers to an incorrect trial. (D) Probability of choosing features from the first correct trial after the rule changes and after one intermediate incorrect trial. In this case, the two non-previous rule features are on the same card. If the two non-previous rule features are on different cards, then the subject chooses them at chance levels.

subject can transition to the exploit state after the second correct trial when there is only one consistent feature between the two correct trials.

Does the previous result depend on the fact that only a single feature is consistently chosen? Let's imagine the same situation but when two features are consistently chosen. E.g. features A and B are chosen on trial $t + N + 1$. Then, on trial $t + N + 2$, they choose feature A. The subject can "stay" and choose feature A again, or they can "switch" to feature B on trial $t + N + 3$. If they are correct/incorrect on trial $t + N + 2$, the best situation is to "stay"/"switch." In Fig. 3.9(B) when trial $t + N + 2$ was correct, the subject usually sticks to feature A, indicating that quickly achieve the exploit state. In Fig. 3.9(C) when trial $t + N + 2$ was incorrect, the subject is able to switch to feature B only when $N \leq 1$. When $N \geq 2$. There is an equal chance of staying vs switching, indicating that do not always transition to the exploit state.

The subject typically transitions to the minor exploit state after the second correct trial, even if there are incorrect trials in between. This is dependent on how many features were correctly chosen however. If multiple features were consistently correct and the chosen feature is incorrect, they need at least one more trial to identify the correct rule.

Transitioning From the Exploit State

The most likely transition out of the major exploit state is the random state. I analyze trials after N correct trials in a row, initialized with an incorrect trial before hand. On the final correct trial, t, the subject chooses features A, B, and C. Feature A is the previous rule feature. After the N correct trials, when the subject incorrectly chooses the rule feature A on trial $t + 1$. How likely are they to choose features B and/or C on trial $t + 2$? While the most common rule change situation is 8 out of 8 correct trials, it's possible for the rule to change after less correct trials due to the 16 out of 20 trials to criterion. In Fig. 3.10(A), the probability of choosing features B and C is above chance when N is small. When N

is large, they are likely perseverating on the previous rule feature. The same trend occurs if features B and C are split between two cards (not shown) on trial $t+2$. This indicates that the subject may switch to the hypothesis testing state after a rule change only if the 16 out of 20 rule change criterion occurs.

It is interesting to observe the perseveration statistics when the subject perseverates and incorrectly chooses the previous rule feature for M trials. With the same initialization as before, how often does the subject to choose the previous rule feature? In Fig. 3.10(B), the probability decays with M. There is distinct behavior beginning at $N = 4$, where the subject perseverates after a single incorrect trial. After 3 incorrect trials, the subject usually converges to chance. This indicates that the subject likely transitions from the minor to major exploit state after 4–5 total correct trials in a row.

Following the 8 out of 8 correct trial criterion, Fig. 3.10(C) shows the probability of perseverating and choosing the previous rule feature after specific sequences of correct and incorrect trials given that the rule feature is chosen for every trial. There are little differences between beginning or not beginning with a correct trial, since this is the same adding a ninth correct trial into the criteria. However, there is a difference when an incorrect trial occurs first. In general, if an incorrect trial occurs and then a correct trial, the subject perseverates for longer.

Finally, can the subject transition to the hypothesis testing state at all from the major exploit state? I analyze the situation when they reach the 8 out of 8 trial criterion, are incorrect on trial t, and correct on trial $t+1$. Stimuli are NOT repeated, and the subject chose features B and C on trial $t+1$. I ignore the previous rule feature. In Fig. 3.10(D), features B and C appear on the same card in trial $t+2$. In this case, the subject focuses on features B and C slightly above chance. If two incorrect trials appear, they do not switch to features B and C. Also, if features B and C appear on two separate cards, the subject does not fixate on them. From this, the subject usually does not switch to the hypothesis testing state, even when given a bit of time to end perseveration. Instead, they perseverate until they reach the random state.

Conclusion of State Analysis

From the analyses in this section, one can form a "state" like description of the subjects behavior (Fig. 3.11). The hypothesis is as follows: the subject begins in the random choice state. Once correct, they transition towards the minor exploit state. If incorrect, they transition towards the hypothesis testing state where they test multiple possible correct features. If two correct trials occur with a single consistent feature, the subject transitions back to the minor exploit state. After about 4-5 total correct trials, they transition towards the major exploit state. At this point, the rule usually changes, and the subject perseverates until they return to the random choice state.

Another natural way to describe these results is one with a sliding scale between the hypothesis testing and major exploit state. More correct trials lead the subject to the major exploit side while more

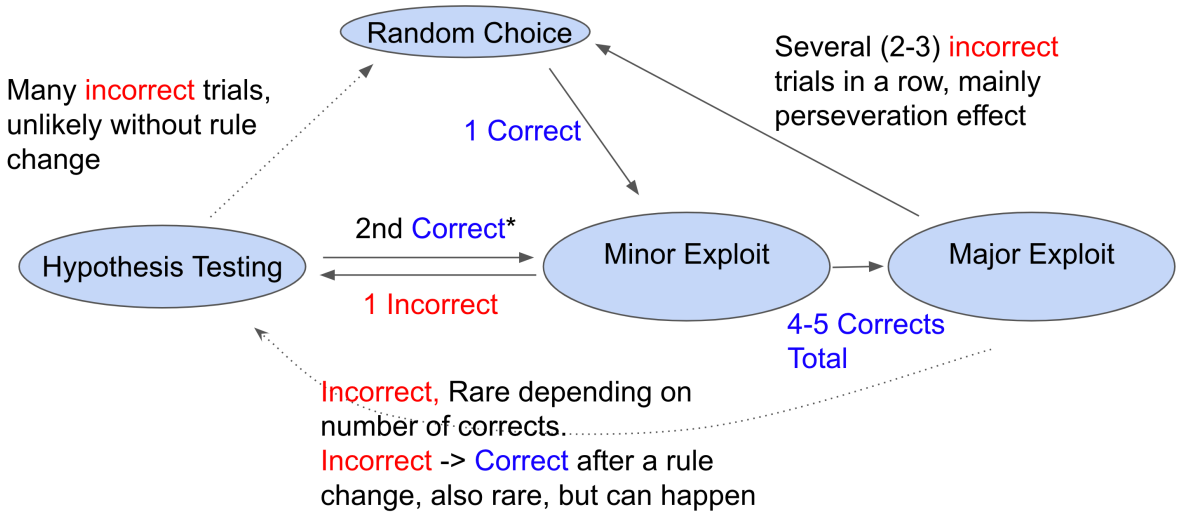

Figure 3.11: Diagram of state description. *If first and second correct share 1 feature. If two features are shared and there are ≤ 1 incorrect intermediate trials, Sam needs 1 more trial (with 1 overlapping feature) to tell which of the shared features is correct

incorrect trials lead to the hypothesis testing state. Either way Fig. 3.11 offers a convenient way to describe the actions of the subject. However, any results should be compared to the actual statistics instead of this description.

These statistics can naturally be explained by the reinforcement learning model in Sec. 4.3.2. When the subject receives correct feedback, they tend to focus on these features more often. A single correct trial when randomly selecting cards leads the subject to fixate on the chosen features. Multiple correct trials leads the subject to begin fixating on a single feature: the rule feature is constantly rewarded and thus the subject weighs this feature more. While exploiting, the subject focuses and perseverates on a single feature. If incorrect, they likely do not transition to the hypothesis testing state, since the majority of the reinforcement learned behavior is focusing on a single feature. Instead, after multiple incorrect trials, all weights decay until the subject is choosing randomly. Altogether, the state-like description of the monkeys behavior agrees with the reinforcement learning computational framework.

3.3 Visual Search Strategy

One topic of interest is how the subject searches the visual space for the target image. How do subjects efficiently search targets, update their beliefs, and decide to continue or stop searching? This can be affected by other correlates, such as motivation or focus on a particular feature. There are many further questions related to the visual search. For example, does the search strategy change throughout the task? Does the subject use peripheral vision to quickly hone into their choice? How does the pre-saccade, where the subject moves their eyes before the objects appear on screen, affect their choice? Is there a bias in choosing stimuli based on position on the screen? This section seeks to answer these broad visual based

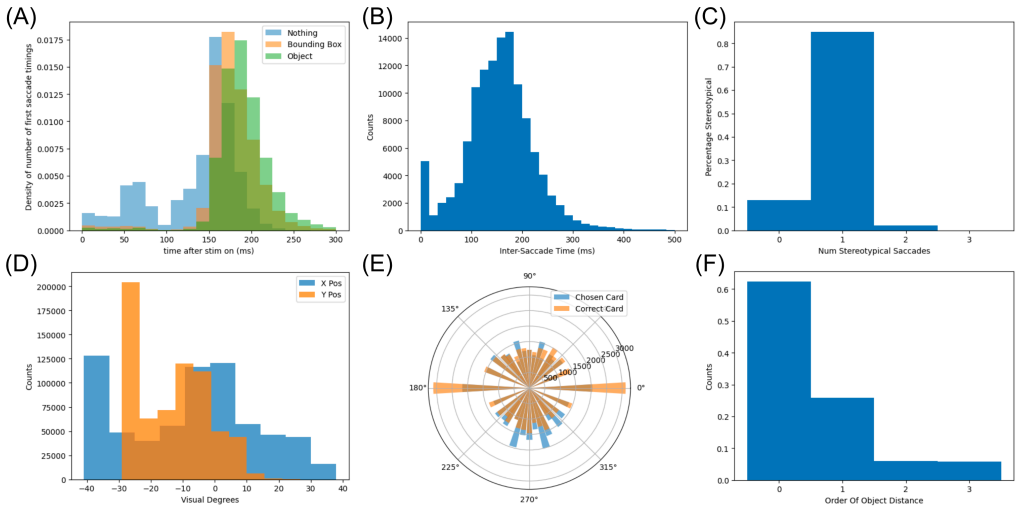

Figure 3.12: General eye movement statistics. (A) Timing of the first non-stereotypical saccade, typically 150-200 ms after simulus onset. (B) Inter-saccade time, estimated as 100-200 ms. (C) Percentage of the number of stereotypical saccades. (D) X and Y positions of the eye at the beginning and end of each saccade during feedback, showing a bias in the negative Y direction. (E) The positions of each chosen card is compared to the positions of each correct card, showing a downward bias and a horizontal inward bias. (F) The percentage of times the subject looks at the closest objects after the objects appear on the screen.

questions.

3.3.1 General Saccade Statistics

General statistics are found for 42 of monkey S's sessions between 2018-06-15 and 2018-10-15. The first question is the timing of the first non-stereotypical saccade after the objects appear on the screen. Stereotypical saccades are eye movements where the subject immediately looks downward, likely without processing any visual information. The timing of the first saccade (Fig. 3.12(A)), divided based on the type of saccade, occurs after around 150-200 ms, consistent with literature [140]. The subject is faster if they look randomly to a blank part of the screen, which is explained by lack of screen processing. There is also a slight decrease in time of about 20 ms when the subject saccades towards a bounding box instead of inside the shape itself. Next, the inter-saccade time interval (Fig. 3.12(B)) is about 100-200 ms, which again aligns with litature [140]. The spike in counts at 0 ms is most likely due to an error in saccade definitions, where a single saccade can be accidentally divided into two consecutive saccades. Meanwhile, a quick investigation into the stereotypical saccades show that the majority (~ 87) of trials have at least one stereotypical saccade (Fig. 3.12(C)). Altogether, these show that stereotypical saccades occur in most trials, and non-stereotypical saccades take around 100–200 ms to process.

Is there a bias in the subjects' choice and preference on the screen? Fig. 3.12(D) shows the position on the screen for the start and end of every saccade during the feedback period, where there is no visual

information. The subject prefers to have an eye position in the bottom half of the screen, as shown in the orange, Y position histogram. There is no bias in the horizontal direction. The peaks at the far left of the plot occur due to tracking the eyes outside the field of view or when the eyes are closed. There is further evidence for a bias downwards in Fig. 3.12(E), which shows the position of all chosen cards compared to all correct cards tends to be downward. There are a discrete number of positions, so every bar on the plot includes a single discrete position as an angle from the center of the screen to the object. There is also a relative horizontal bias inward. The downward bias in Fig. 3.12(D,E) aligns with the downward stereotypical saccade. It's possible that this bias is due to the experimental design, where the subject's head may be tilted higher than the desired position[1].

Finally, one hypothesis is that the subject prefers to saccade towards close objects. For the first non-stereotypical saccade after the objects appear on the screen, the objects can be sorted based on distance from the current eye position. The percentage of times where the subject saccades to each relative position, ignoring saccades to a blank screen, is shown in Fig. 3.12(F). 60 percent of the time, the subject saccades to the closest object, and around 26 percent of the time the subject saccades to the second closest object. This provides evidence against a strong use of peripheral vision, since peripheral vision would lead to a more equal distribution across object distances. However, there is a non-zero probability of saccading towards the furthest two object. Either the subject randomly wants to look across the screen. Or the subject saw a far away object that matched their belief, and the subject decided to saccade directly to that object.

3.3.2 Saccade Strategies Statistics

Different possibilities exist for the specific saccade strategies used searching the visual field, such as examining all four objects before selecting a card, using peripheral vision to look at only one card right away, or returning to a previously observed card. This section asks focuses on statistics of the preferred strategies.

Strategy Labels

I define common "saccade strategies" that the subject typically uses (Fig. 3.13). In the first group of strategies, which I term "ring around the rosie" or R1, R2, etc., the subject looks at N different objects in a row. R1-R4 is defined as saccading to one, two, three, or four different objects. R5 is defined by saccading to all four objects and then saccading to any of the objects. Other possible strategies occur in an atypical order (Fig. 3.13(B)), and they are labeled in the pattern that they appear in. E.g. ABA is when the subject looks at card A, then card B, and back to card A as their final choice. For completeness, two other groups of strategies are listed. First, any strategy with at least 6 searched objects is listed as

[1] This idea came from Patrick Zhang in discussions on the location bias of the subject.

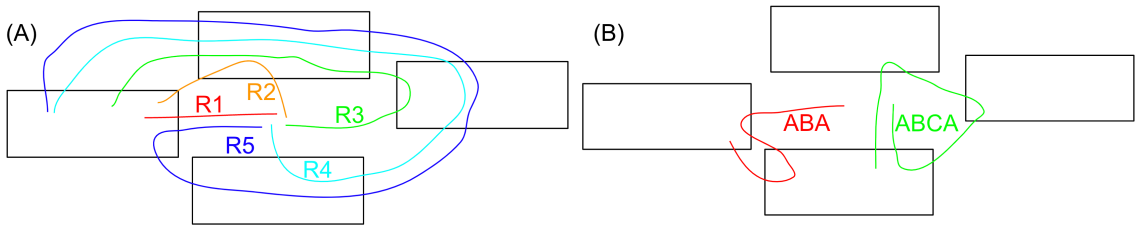

Figure 3.13: Depictions of key "saccade strategies." (A) Ring around the rosie strategies, where the subject looks at N number of different objects in a row, not necessarily in a circular fashion. R5 occurs by looking at four different objects and returning to any object for the 5th saccade. (B) Other strategies, labeled by their atypical object orders.

">5". Any strategy where the subject looks to an empty part of the screen are listed as E2, E3, E4, or E5. E is used to indicate an error, where this may occur due to a calibration error or an uncommon strategy.

Preprocessing

The behavioral data from all of Monkey S's sessions are used. The sessions are divided into 4 groups, designated in Sec. 2.1. The first group is the normal task, and what is typically used in neural analysis. The second group is when the timeout period was significantly shortened. The third group is when alternative stimuli were presented. The fourth group is the original stimuli but with a shortened timeout period.

The eye data is used, as is described in Sec. 2.9, to define when the subject is looking at particular stimuli. Their eye must be in the area of the bounding box. Stereotypical saccades are NOT included. Microsaccades, where the eye moves within the bounding box of a single object, are ignored.

Saccade Strategies for All Sessions

Across all sessions, the percentages of each strategy used are shown in Fig. 3.14(A). The most common strategies involve low numbers of saccades. Particularly, 30 percent of the time the subject will saccade to a single object. 26 percent of the time, the subject will look at two objects. 15 percent of the time, the subject will look at three different objects. The most common group of strategies are the ring around the rosie strategies ("R4" for example). It is common to look at all four objects (either R4 or R5), indicating a preference for new information. It may also indicate that the subject stops abruptly if they see a desired feature instead of looking further. This is consistent with the idea that the subject chooses features to look for before seeing the visual information.

There is also a prevalence of strategies involving the return to an earlier object (Fig. 3.14(A)), such

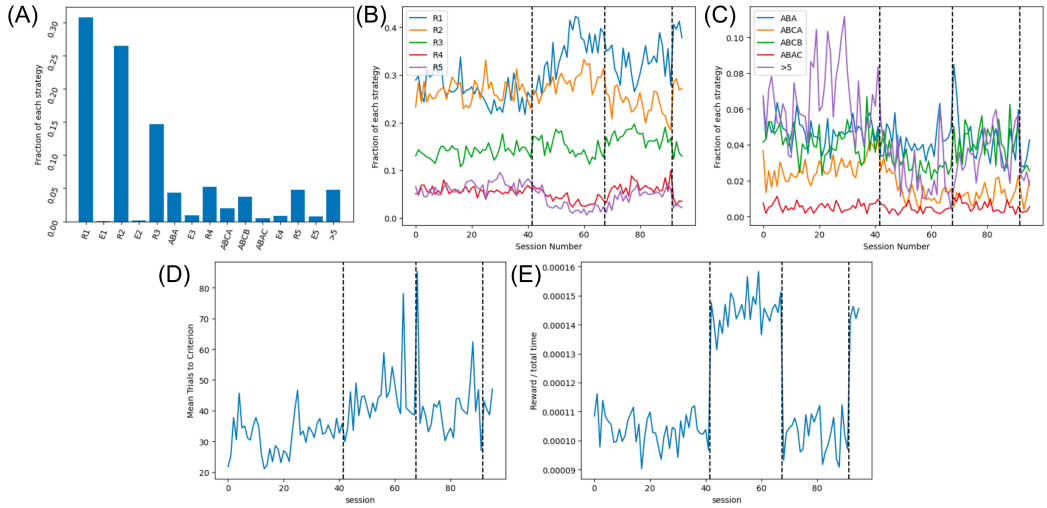

Figure 3.14: Session level saccade statistics. (A) Fraction of common strategies, summed across all sessions. There is a preference for strategies with less searching and more commonly without observing the same card twice. For (B-E), black dotted lines separate groups of sessions; the second and fourth group of sessions have a short timeout period and the third group of sessions have different stimuli. Sessions 2018-11-07 and 2018-12-04 were dropped due to major outliers in percentages. (B, C) Fraction of strategies across sessions. There is a shift in strategies after the second group of sessions, where there is a high preference for R1 strategies. (D) Average number of trials to criterion, verifying that the subject started solving the task slower. (E) Total reward per time spent, showing an increase in reward when the timeout period was shortened.

as in ABA, ABCB, and ABCA. They typically only look at a single other card before returning to a choice. Further, the low probability of the ABAC strategy indicates that the subject usually does not double-check and avoid a card. Instead, if returning to a card, the subject is likely going to choose it.

There are relatively small likelihoods of error strategies (e.g. E2) in Fig. 3.14(A), which occur when the subject looks to an empty region of the screen. This helps confirm the initial calibration of the eye data, since a bad calibration would lead to higher percentages of these strategies. However, since the percentage in E1, where the subject saccades to nothing, is non-zero, the calibration is not perfect.

Saccade Strategies at the Session Level

The percentages are analyzed further by separating based on session. The relative percentages of each strategy per session (Fig. 3.14(B,C)) show that a change in strategy particularly when the timeout period shortens. During this time, the subject searches less objects: the percentage of R1 and R2 increase while ABCA, ABCB, and >5 all decrease. When the timeout period is lengthened in the third group, the percentage of R1, R2, and >5 never return to the original percentages in the first group, implying that the timeout period altered the main strategy of the subject.

The drastic change in saccade strategy percentage is likely due to the spontaneous decrease in time out periods when incorrect. If there is a large timeout period, a subject is encouraged to perform better than if there is a small timeout period. Given the increase in R1 and R2 strategies, the subject may be rapidly choosing cards, leading to a worse performance. As seen in Fig. 3.14(D), the average trials to

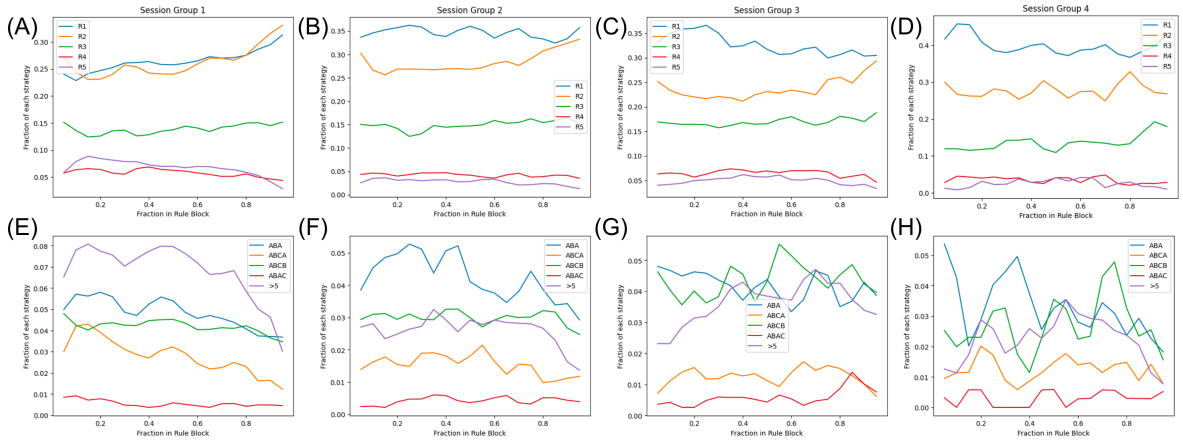

Figure 3.15: Percentage of common strategies per session groups (columns) and fraction in rule blocks (x-axis). Rule blocks are partitioned into windows of 10 percent of the total block length. The increase of R1/R2 (A) and decrease of >5/ABA/ABCA (E) indicate change of strategy throughout the rule block in first session group. R1 is used often in other session groups (B-D), likely indicating rapid card choice.

criterion increases after the short timeout period is introduced, leading to a worse performance. It may be possible, however, that they still obtain a large amount of reward. The reward per total time spent in the task is shown in Fig. 3.14(E), where a dramatic increase in reward rate occurs after the short timeout period is introduced. Thus, it's likely that the subject realized that worse performance still leads to large rewards. Further, the subject doesn't return to the original behavior when the timeout period is restored, likely indicating a permanent strategy alteration.

Saccade Strategies at the Rule Block Level

Are there differences in saccade strategies based on the trial in the rule block? The data are binned in terms of their fraction of total trials in each block; the bin size is 10 percent of trials, and the stride is 5 percent of trials. Due to perseveration, there may be similar behavior at the very beginning and end of each rule block. The relative percentages of each strategy are shown in Fig. 3.15. There are noticeable differences in saccade strategies for each group of sessions, which is expected based on the differences throughout the sessions (Fig. 3.14). In the first group of sessions, there is a major difference between the beginning and end of the rule block. R1 and R2 strategies occur more often throughout the rule block. R4, R5, ABCA, and >5 strategies occur less often as a function of percentage in rule block. Towards the beginning of the rule block, they are likely learning the rule and need to search more objects to make a choice. Later, once they have an idea of the rule feature, they saccade to less objects and choose quickly.

There is a stark difference in behavior for the second group of sessions onward, when the timeout period is shortened (Fig. 3.15(B,F)). There is less of an effect of changing the saccade strategy as a function of percentage in rule block, with the exception of a minor increase in R2 and decrease in the ABA and >5 strategies. Also, the percentage of R1 is consistently higher and stable throughout the rule

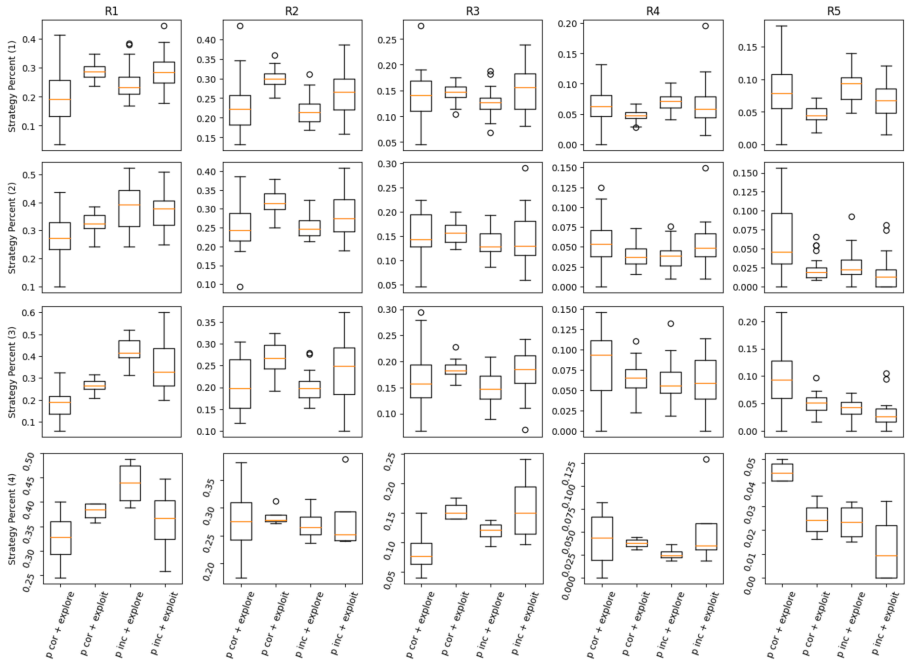

Figure 3.16: For each session group (rows, increasing downwards), fraction of R1-R5 strategies used (columns) depending previous trial feedback and whether the subject is in an "explore" or "exploit" state. For the first group of sessions, R1 typically is high when the subject is exploiting. For the rest of the sessions, R1 is high when previously incorrect and exploring, indicating a permanent saccade strategy switch. For the 2nd through 4th session groups, R5 is typically large when previously correct and exploring, indicating a shift to searching for features after the first correct trial in a sequence.

block. This provides evidence for the hypothesis that the subject permanently alters their strategy to one that emphasizes fast choices.

Saccade Strategies at the Trial Level

Finally, one may be interested at the saccade strategies used depending on the current sequence of trials. One may expect more exploratory behavior if the previous trial was incorrect for example. One can also define an explore vs exploit state, when the subject is typically exploring or exploiting. This analysis was completed before more recent behavioral work, which may be more applicable to finding specific internal beliefs.

The explore and exploit state is defined based on the general idea that a subject typically explores when receiving incorrect feedback and exploits when receiving correct feedback. The state switches from explore to exploit after 2 correct trials in a row, and the state switches to explore after 2 incorrect trials in a row. The intuition is as follows: if the subject is exploring and got a trial correct, they likely do not understand the rule and continue to explore. After two correct trials in a row, they likely believe they understand the rule and switch to an exploit state. If incorrect while in an exploit state, they tend to perseverate and maintain an exploit state. After two incorrect trials, the subject may stop perseverating and are now in an explore state. The state is initialized to explore if the first trial was incorrect and exploit if the first trial was correct. As an example, if the feedback sequence is correct, incorrect, and

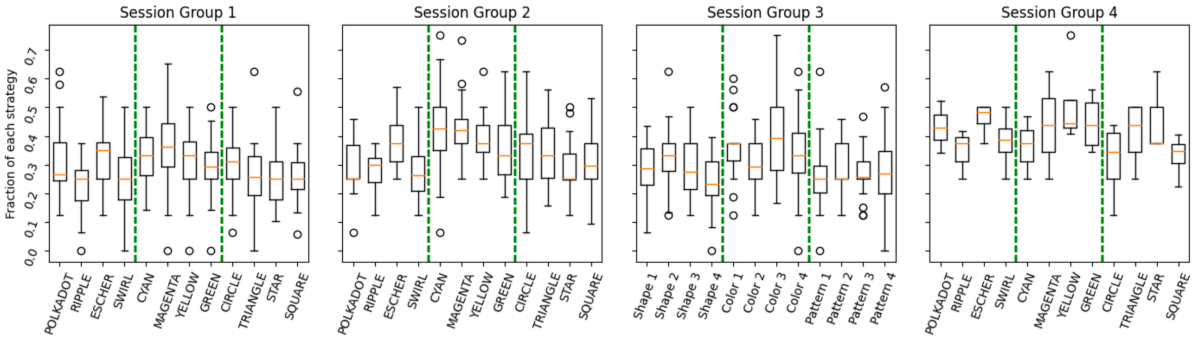

Figure 3.17: Fraction of R1 strategy during last 8 out of 8 or 16 out of 20 trials for each feature for different session groups. For the third session group, there is a mixture of different stimuli, so they are generically labeled. In general, the fraction is high for color rules and some specific features, such as the escher pattern. The overall scale of 0.3 to 0.4 indicates that peripheral vision is barely used to identify the card of choice.

incorrect, then the state sequence would be exploit, exploit, and explore. While simple, this model provides an easy way to classify trials as explore or exploit.

Focusing on the R1-R5 strategies, the fraction of each strategy is computed depending on the previous feedback and the current internal state in Fig. 3.16. In the first group of sessions, R1 is particularly high when in an exploit state, which could be due to focus on a particular rule and directly saccading to it. Afterwards, R1 is typically high when previously incorrect and exploring. In the second group of sessions, R1 is also high when previously incorrect and exploiting. This highlights a major difference between the first group of sessions and the rest: in the rest of the sessions, when the subject does not have an idea of the rule, they saccade quickly. Intuitively, they do not waste time searching while the rule is unknown. Similar behavior is found across all sessions in R2 and R3 when exploiting, and R2 and R3 strategies are used more when previously correct. This is sensible when quickly searching for the rule without one shot saccading.

In the first group of sessions, there is a relatively similar use of the R5 strategy when not exploiting and previously correct. In all of the other sessions, the percentage is highest when previously correct and exploring. Thus, the subject likely focuses on the previous correct information when it's novel. When this occurs, they may search for a long time to verify what they are searching for. One hypothesis is that the subject, during the second group of sessions onward, rapidly saccades when they are unsure. Once they are correct, they switch to a strategy where they search for a long time and remember what was previously correct. This implies that the subject computes information and belief of the rule while searching the visual space. If the subject computed belief of the rule earlier on, they most likely wouldn't look at 5 objects while deciding the card to choose.

Easier to Use Peripheral Vision for Color

Does the subject use peripheral vision to make choices? Vision science would predict that color may be more discriminable in peripheral vision than pattern. To investigate this, only the last 8 out of 8 or 16

out of 20 correct trials per block are analyzed. Then, the percentage of R1 strategies compared to all strategies are computed. In this subset of trials and strategies, the subject may use peripheral vision to quickly make a choice. The percentages are computed for each feature and session group in Fig. 3.17. For all cases, there is a preferential bias toward the color rules, indicating that colors are easiest to find with peripheral vision. Interestingly, it appears that the Escher pattern is consistently easier to find, indicating that some features may be easier to find than others.

From Fig. 3.17, the scale of all percentages is around 0.3 to 0.4, nowhere near high enough for consistent use of peripheral vision. However, it is higher than chance of 0.25, where lack of peripheral vision will lead to a quarter of saccades directly to the rule. Overall, this shows that the subject can use peripheral vision to make a selection quickly, but they do not often do so.

No Predictability in Neural Data

One minor question is whether there is predictability of the various saccade strategies in the neural data. Given blockwise differences in saccade strategy, it is possible that the strategies themselves are indicative of confidence. E.g. later in each block, the subject is more likely to saccade directly towards their target with the "R1" strategy. If this is true, perhaps there is a difference in saccade strategy during the cross fixation period, where they may plan out their action before the objects appear on the screen.

I focus on the difference between the "R1" strategy and strategies with 4 or more saccades for session 2018-08-02 ($>$ 300 trials per condition). Pre-processing wise, the bipole data is normalized, notch-filtered around the 60 Hz harmonics, bandpassed to different frequency ranges, Hilbert transformed. The absolute value of the Hilbert transform shows the power in the chosen frequency band for each trial. The power is averaged for 100 ms bins with a 50 ms stride during the cross fixation period. 20 independent linear decoders ($C = 10^{-7}$, dual = False) for each time point are trained on balanced data with a train/test split of 80/20. They are compared to the performance on shuffled labels. For the raw LFP and chosen frequency bands, there is little to no decodability of the distinct saccade strategies (not shown), suggesting that the LFP does not encode any preference towards the R1 or ≥ 4 saccade strategies. It's possible that this information does not exist at this time as well. Instead, given the decodability of various saccade statistics in Sec. 7.3.5, the subject may search more randomly.

3.4 Visualization from Spectrograms

A useful first analyses is to examine the spectrograms of sample channels to obtain intuition about possible frequency bands of interest. The frequency bands with interesting behavior should be similar to those found in the bandpassing preprocessing step (Sec. 2.7) and to those defined in literature [104]. This section will focus on a few interesting spectrograms and average spectrograms over relevant behavior.

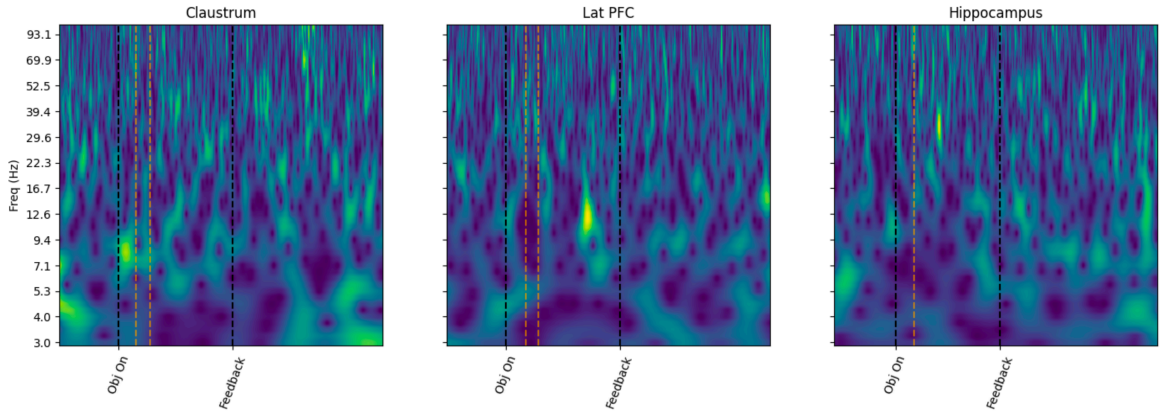

Figure 3.18: Example spectrograms for individual trials, starting from the cross fixation until 1500 ms after feedback. Black dotted lines indicate major changes in the task. Orange dotted lines are saccades to new objects. There is a burst in activity at 8 Hz in the claustrum after the objects turn on and before any saccades. When fixating on the chosen object, there are bursts in power at 10 Hz in the lateral PFC and at 30 Hz in the hippocampus.

Spectrograms are calculated with the Morlet Wavelet Transform (MWT), described in Sec. 2.5. The spectrum is computed for 150 trials, z-scored for every frequency band, and separated into spectrograms per trial. Spectrograms are found for selected bipoles in monkey S's session 2018-08-02, after the preprocessing (Sec. 2.8) and removal of 60 Hz noise. Different bipoles are shown for every figure. Fig. 3.18 shows three example channels spectrograms for random trials. There is a substantial amount of activity or noise driving the system. Despite this, there are bursts of power in each channel. In the claustrum, there is an increase in power at 8 Hz after the objects turn on but before the subject saccades to any object. In the lateral PFC and Hippocampus, there is an increase in power while fixating on the choice.

There are several ways to increase the signal. Classically, averaging across trials can give task specific variables. One could also look at bursts of activity in the LFP. During bursts, the LFP power jumps above the normal level of activity, which can be useful for filtering out noise.

After aligning the trials to cross fixation and object fixation, the power is averaged across the first 150 trials to get a mean spectrogram. Example mean spectrograms are shown in Fig. 3.19, where task specific variables are observed. In the amygdala, there is large power at all frequencies before fixating on the cross. In the dorsolateral PFC and striatum, there are increases in power during the feedback period. These indicate that activity patterns occur at specific times during the task. While only a few shown here, many bipoles have interesting behavior.

One downside with mean spectrograms, however, is that information can be lost. If, for example, there was a large response that occurs within a half second window, it would appear as a long/large response. This may be misleading when the power only increases for short time periods, which appears to be the case based on Fig. 3.18. Other information can be difficult to observe, such as the search period, where saccades can occur at anytime. Although, in this case, one could align to individual saccades.

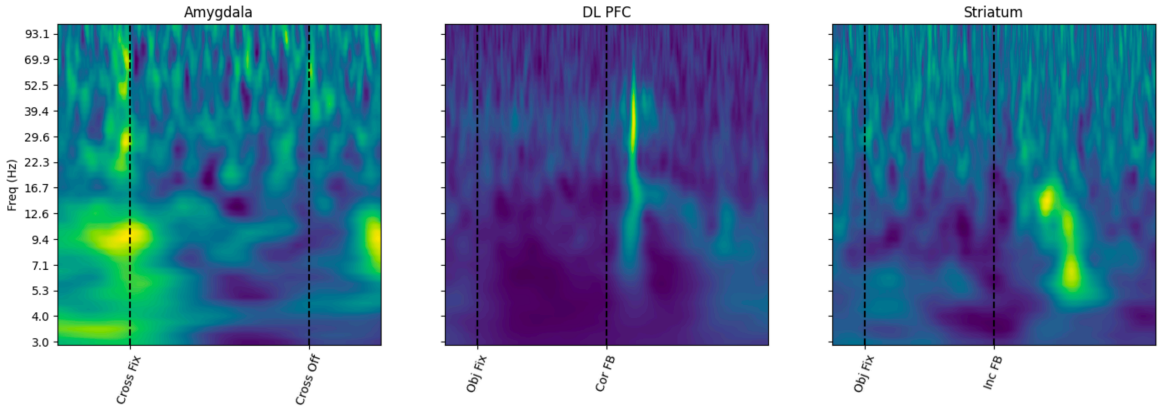

Figure 3.19: Example mean spectrograms during both the cross fixation and during the feedback period. Spectrograms are only averaged during correct or incorrect trials. Consistent activity is shown across trials. In the amygdala, there is consistently high power immediately before the fixation on the cross. In the dorsolateral PFC and striatum, there is consistent activity in different frequency bands at different times during the feedback period.

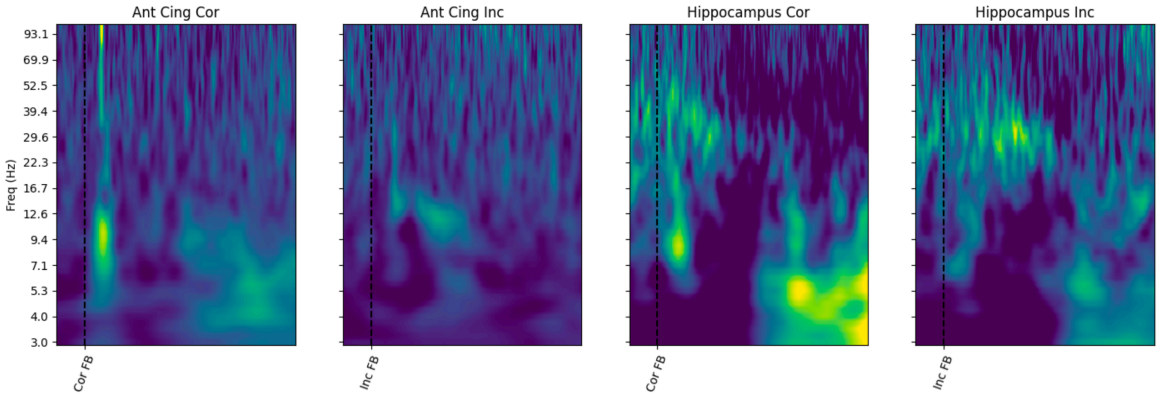

Figure 3.20: Example mean spectrograms for correct and incorrect trials for two signals. Black lines occur when the subject receives correct (cor) or incorrect (inc) feedback (FB). In the anterior cingulate, there is a power increase at 10 Hz only during correct trials. There is also a high frequency power increase only during correct trials. There is an increase in power in incorrect trials at around 12 Hz. In the hippocampus, there are numerous differences below 12 Hz between correct and incorrect trials. There is also a difference at around 30 Hz, where there is a larger increase in power during incorrect trials.

Spectrograms Show Feedback Discrepancy

There are many bipoles that show feedback differences. Example differences are shown in Fig. 3.20. In the anterior cingulate, there are differences in power immediately after feedback is received. In the hippocampus, there is an increase in power at 30 Hz for incorrect trials, among other differences. The differences indicate that different neural processes occur during correct and incorrect feedback. There are a large number of bipoles that have feedback differences in the spectrograms all across the brain. Further analysis of feedback is explored in Chapter 6.

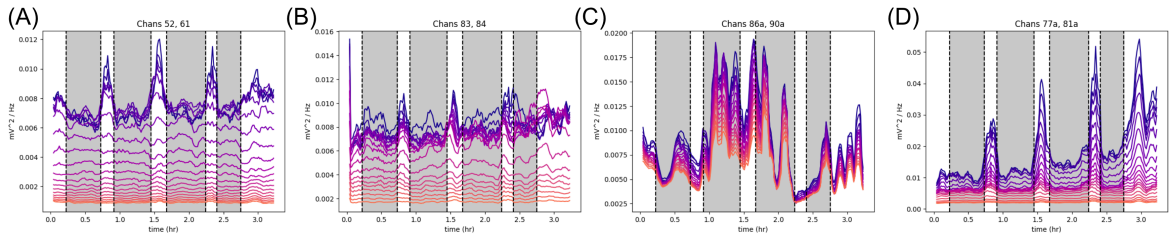

Figure 3.21: Example Morlet Wavelet Transform power for frequencies between 3 (purple) and 25 (orange) Hz for session 2018-09-25. Grey shaded regions indicate when the subject is performing the task. Outside the grey region, the subject is resting. The majority of channels in this session have an increase in power, especially is lower frequencies, when not performing the task. Some channels (C) have large amplitude fluctuations throughout the session. There are sizable amplitude fluctuations during the task in (A, B, and D).

3.5 Power Throughout Sessions

One concern in time-series analysis is the stability of the signal across time. Ideally, fluctuations are solely due to behavioral correlates. Otherwise, spurious, or nonsense correlations [141], can occur, such as correlations between neural signals and crypto prices [142]. This can occur due to autocorrelation. In typical statistical analyses, points are considered to be statistically independent. Due to autocorrelation, the data are not independent from each other. This is especially concerning in this dataset, since there are expected signal changes at the rule block scale, which is a scale that may include autocorrelation effects. Without caution, one may find correlations where no such correlations exist.

There are several reasons why the signal may not be stable. The subject may undergo state changes due to motivation, hunger, or arousal, which have been shown to correlate with broad changes in the LFP [143]. Or, the electrodes may drift slightly across the sessions, leading to slight differences in the overall power.

Example bipoles are checked for stability in monkey S's session on 2018-09-25. For individual bipoles, the data is normalized and notch filtered, and the Morlet Wavelet Transform (MWT) is computed for frequencies between 3 and 25 Hz. The amplitude of the MWT is squared to get the power, which is averaged over every 5 minutes with a stride of 1 minute. The resulting slow power drift (Fig. 3.21) shows fluctuations in the power while performing the task. For example, in Fig. 3.21(D), the power is noticeably different in the different shaded regions. Fig. 3.21(C) shows an even more problematic bipole, where the power shifts at seemingly random times. Care is needed when correlating this LFP data to behavior, especially at the rule block scale. Rules change about every 30 trials, or every few minutes, which is consistent with fluctuations in Fig. 3.21.

Fig. 3.21 also shows an interesting phenomenon during rest periods. Many bipoles have a spike in power, especially in low frequencies, during the rest periods of the task. Boosts in power may occur during sleep periods, which are known to have high power in the LFP in low frequencies [144].

3.6 Demixed Principal Component Analysis

One natural method for understanding behavioral correlates is to search for low dimensional representations that explain the neural data. Demixed Principal Component Analysis (dPCA) [145] seeks to explain the most amount of variance possible in the data while simultaneously separating out trial conditions. dPCA works by first averaging across different marginalizations: e.g. across time, feedback, choice, etc. It then finds low-dimensional representations in each marginalization.

Many trial-wise conditions can be applied. Unfortunately, if every possible condition is included, there will few trials and poor statistics. For this reason, only a few marginalizations are considered. The first marginalization is over feedback and previous feedback, providing a baseline feedback representation. Trials are then divided based on chosen features, allowing for representations of working memory.

The pre-processing is as follows. For Session 2018-08-02, the spiking data (42 neurons) were binned in 100 ms bins with a 50 ms stride around the choice and feedback period. Trials were limited to after the second block, and only trials with "clean" LFP signal are used. If an artifact occurs in the LFP data, the relative spiking data is thrown out due to the fact that spiking is impossible if the LFP has reached the limit of the electrodes measurement range. Trials are included only if both the current and previous trial were correct or incorrect, eliminating any trial where the previous trial was not completed. The data is concatenated across trials and time, z-scored across the concatenated dimension, and reshaped back into the original shape. Finally, the data is averaged across trial conditions, and dPCA can be implemented. Whenever plotted, I follow Kobak et al. [145] and combine the marginalization and the interaction term between the marginalization and time component. In all cases, the explained variance decays in the same manner between PCA and dPCA, validating dPCA.

Feedback Modes

I divide every trial into one of four conditions depending on the current and previous feedback. If both trials were incorrect (347 trials), the subject likely does not know the rule. If both trials were correct (551 trials), the subject likely does know the rule or just figured it out. These two cases can be thought of as consistent or "unsurprising" trials, since the subject likely is not surprised by this feedback. If the subject is correct followed by incorrect (250 trials), they are likely surprised that they got the trial wrong. If they are incorrect followed by correct (250 trials), they are likely surprised that they got the trial correct.

Running dPCA on the feedback and previous feedback conditions leads to some differences in the spiking data (Fig. 3.22). There are distinct differences in timings, where some modes occur early while others occur late after feedback. There is also one mode that switches direction at around 400-500 ms

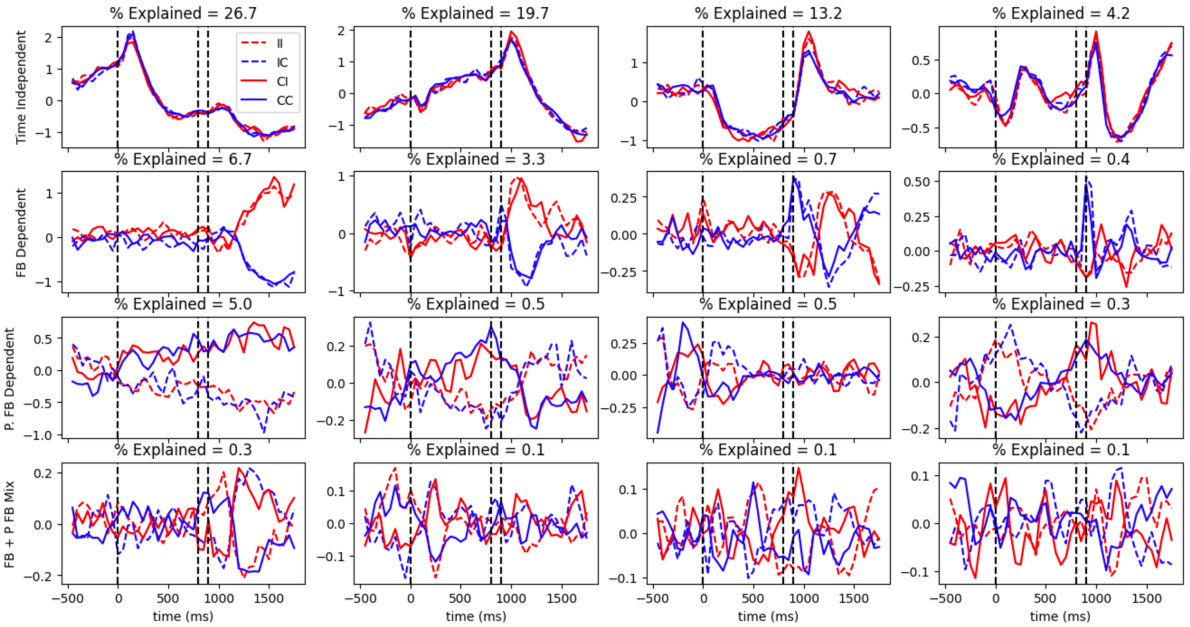

Figure 3.22: dPCA modes for the current and previous feedback. Red/blue indicates the current trial was incorrect/correct. Dotted/solid indicates the previous trial was incorrect/correct. The legend indicates the order of feedback for the two trials (I = incorrect, C = correct). Each row is a different marginalization. Each column is a different mode number, where the first column has the highest explained variance. The top row are condition-independent modes, where there is little difference between feedback. The middle top mode (current feedback dependent) shows differences after feedback begins. The middle bottom mode (previous feedback dependent) shows previous feedback differences during all time periods. The bottom mode (mixing between current and previous feedback) shows differences at all time periods. However, there are only differences between consistent (cor $\rightarrow$ cor or inc $\rightarrow$ inc) and inconsistent (cor $\rightarrow$ inc or inc $\rightarrow$ cor) trials.

after feedback. There are surprisingly differences in the previous feedback modes at all time periods. For the modes during the choice and search period, the previous feedback differences may occur due to a memory of the previous trial. During the current feedback period, perhaps the previous and current feedback are integrated to update the internal model. The cross term between the current feedback and previous feedback shows interesting differences. First, there is a difference in the cross terms during the current trials feedback period. There is also a difference during the choice and pre-choice period. There is a difference in the cross terms when the subject is "surprised" compared to "unsurprised." While it's unclear whether this effect is limited to the previous trial or generally represents confidence, it shows that metrics like reward prediction error (RPE) could lead to understanding of feedback differences.

While not shown, I perform the same dPCA analysis on LFP power in various frequency ranges. Similar effects are seen in the 26-34, 65-87, and 120-148 Hz frequency bands, showing that both the LFP and spiking data are influenced by feedback.

Single Feature Modes

The difference in signal for each feature can show representations of working memory. Each trial, the subject simultaneously chooses three features, and the trial is assigned to each of the three features.

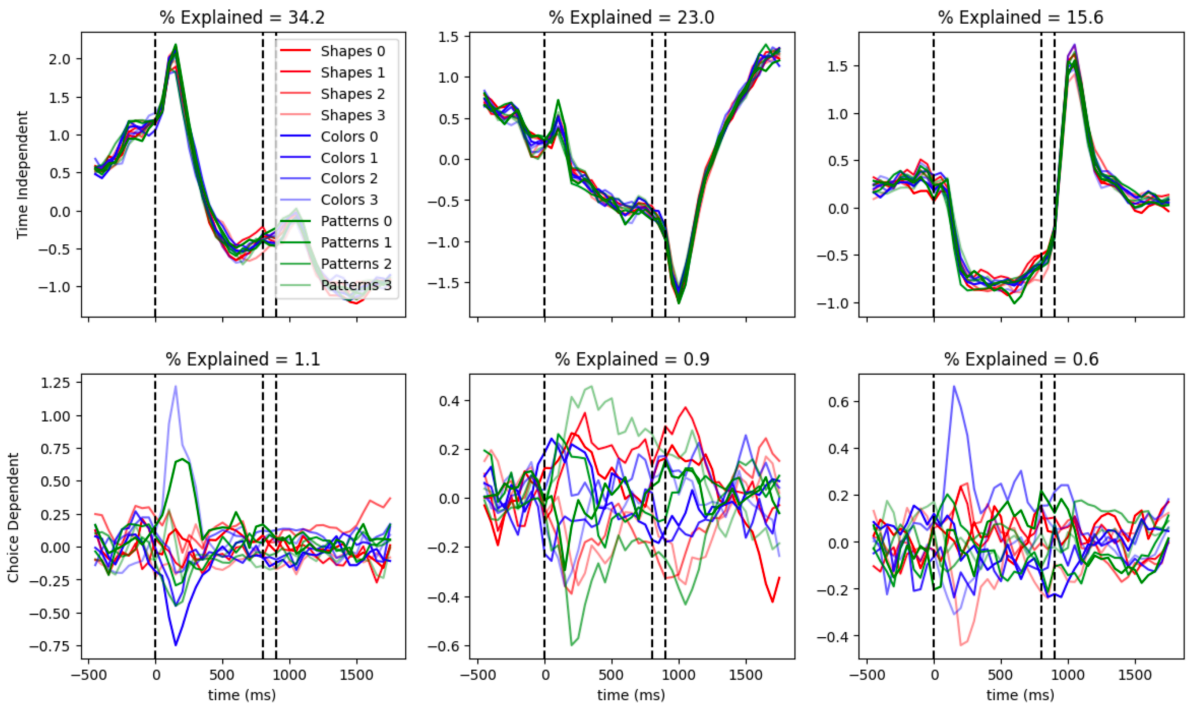

Figure 3.23: dPCA modes for single feature modes. All trials with a particular feature (e.g. color 0) are averaged together. This means that each trial is overcounted three times. The top row shows the condition independent mode. The bottom mode, the feature dependent mode or choice of each feature mode, shows differences among the different features. The modes separate during the the choice and feedback period.

While all trials are overcounted, dPCA uses the average trace for each feature. There are 349 ± 28 trials per feature.

In Fig. 3.23, the signal separates based on features. While expected, the signal doesn't separate based on rule dimension. Instead, random features are separated. Timing wise, the signal is most separable during the choice period. However, there is an ability to separate the features during the feedback period. This indicates that their are feature representations while choosing the card and while remembering the card during feedback. Given that, this representation may be purely due to the choice without regard to the belief of the rule. Since the explained variance is small, very little of the spiking activity is explained by the features chosen.

The same analysis is rand for power in different LFP bands: 26-34, 65-87, and 120-148 Hz. There was no separation between the different features (not shown). Instead, the LFP is dominated by noise. It is likely that feature information is better represented in the spiking data than LFP data.

Single Feature Modes Per Card

While the previous section analyzed the dPCA modes by averaging across all trials for a particular feature, this section considers all feature combinations: every trial belongs to one of 64 possible three feature combinations. There are 22 ± 6 trials per card, leading to worse statistics. In Fig. 3.24, the dPCA modes separate based on feature dimension. That is, for the shape dependent modes, there are

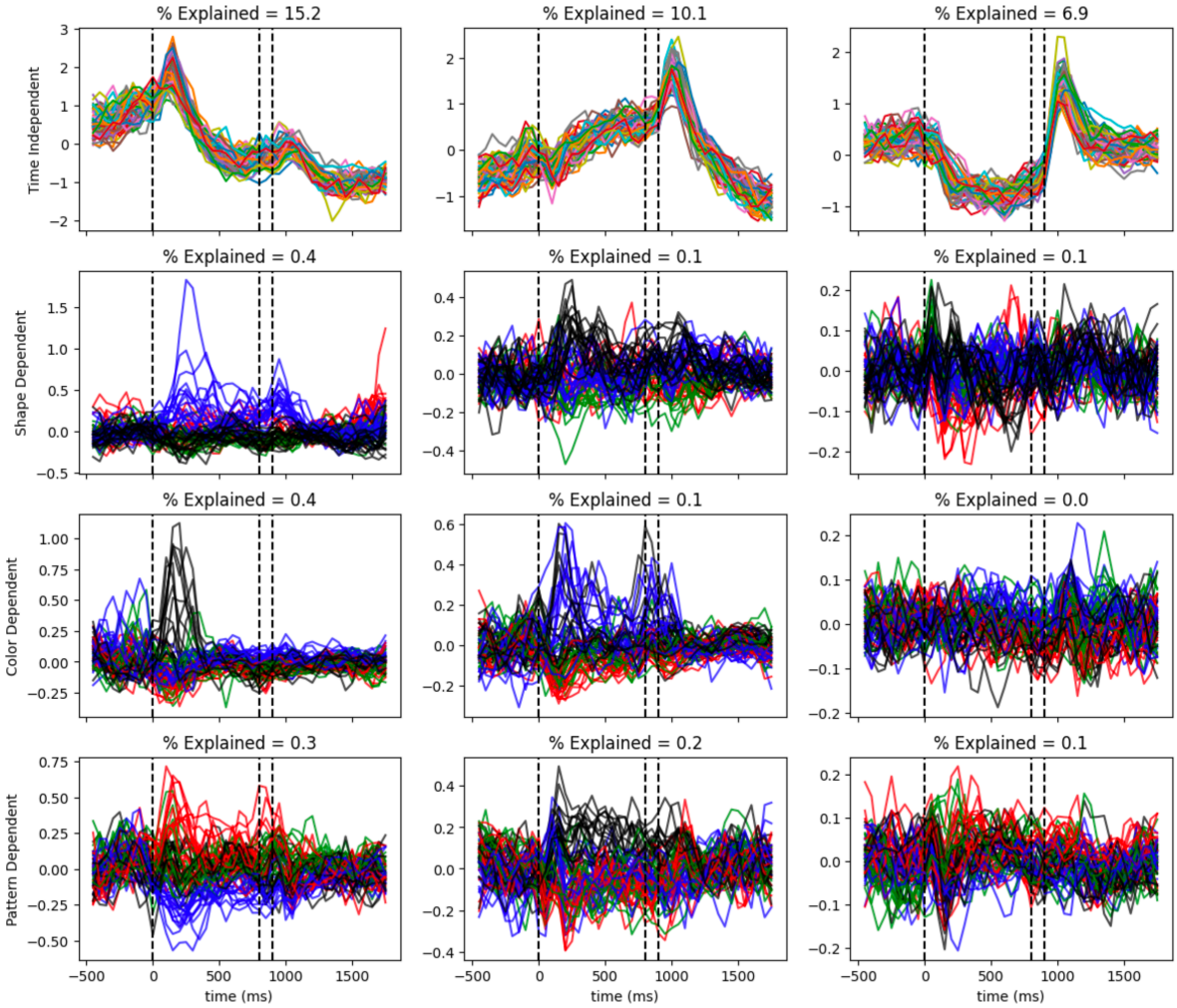

Figure 3.24: dPCA modes for each feature when considered a different dimension. The top row shows that all conditions have similar condition independent modes. For each row below, corresponding to the shape, color, and pattern, the colors are are grouped by the relevant dimension. E.g., for shapes, blue corresponds to shape 1, red for shape 2, green for shape 3, and black for shape 4. The colors do not correlate across rows. Each trace is one of the 64 possible combinations of three features. The different dimensions separate in the choice and feedback period.

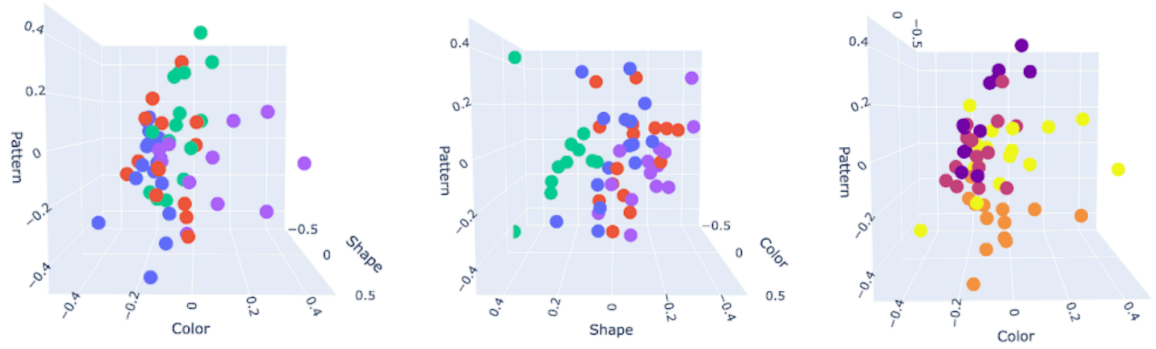

Figure 3.25: Projection of every average trace into the low-dimensional dPCA space at 200 ms into the choice period. The three axes are from the first mode of the color, shape, and pattern dimensions. Colors are distinct for the different colors (left panel), shapes (middle panel), and patterns (right panel). While not perfect, the features do separate in this space.

traces that separate for each shape. The strongest effect occurs in the pattern dependent modes, where multiple patterns are able to be separated. Like in the previous section, the majority of separation occurs during the choice period. Some separation occurs during the feedback period. These modes likely explain the working memory of different features, while they may not explain the belief of any particular rule. The explained variance is small, so the spiking activity is only marginally explained by the different features.

The same analysis is ran with the frequency bands: 26-34, 65-87, and 120-148 Hz. While faint, there is a little bit of separability of shape throughout the choice period in the 120-148 Hz band. This may occur since the high gamma range is known to be correlated to spiking activity [146]. Despite this, the representation is clearer in the spiking data.

I then project the data into a common space and check to see if all 64 traces occur in a common framework in the low dimensional space. The largest separation occurs at 200 ms in Fig. 3.24. The data at 200 ms is projected into the first mode for each dimension in Fig. 3.25, where the data separates. Unfortunately, the representation is not uniform. The lack of separation most likely originates from the lack of trials. It would be interesting to try a similar analysis by bootstrapping the data across all sessions. Using an approach with more data may lead to better separation and more uniform low dimensional representations of each card. Despite this, it's interesting that there is a separation of the different features, indicating that the features of each card may be represented in a low dimensional way in the subjects mind.

Chapter 4

Behavioral Models & Species Strategies

To understand the underlying representations involved in solving this task, one needs behavioral modeling. In this task, it's extremely difficult to understand why the subject made their particular choice. For example, if the subject chose a cyan triangle with an escher pattern, did they choose the card because it was cyan, a triangle, because it had the escher pattern, or because of a combination of these three features? While difficult, one may assume a simple behavioral model where the subject focuses on the rule feature during the last 8 out of 8 or 16 out of 20 correct trials, a problematic assumption. First, this severely limits the number of trials that can be used. Second, it is unclear when the subject begins to focus on the rule feature. Third, the representation may not have a consistent presence in the neural data, where the representation may only be salient when the subject is exploring. Lastly, the subject becomes distracted by alternative features to the rule during the 16 out of 20 trial criterion (Sec. 3.2.2), implying that the rule representation is unreliable.

With these problems in mind, behavioral modeling can lead to more reliably predictions of the subjects choices on EVERY trial. This work, and the remainder of this chapter, is in the process of being published [2] with co-authors: Patrick Q. Zhang, Po-Chen Kuo, Michael J. Jutras, Bhargavi Ram, Jack Lin, Robert Knight, Elizabeth A. Buffalo, Edgar Y. Walker, and Adrienne L. Fairhall. This work focuses on both monkeys and humans, where different behavioral strategies are found across species. In this chapter, the task is referred to as a multidimensional choice task to avoid any confusion with the original implementation of the WCST [95].

4.1 Abstract

Identifying the features of complex observation spaces that are relevant to decisions is necessary to successfully navigate intricate environments. Frequently, credit cannot be uniquely assigned to any one particular action, and it may be computationally challenging to retain a memory of all possible action-

observation pairs. Inter-species comparisons during high-dimensional credit assignment problems are key in understanding species specific or common computational frameworks that lead to varying levels of proficiency. Here, to reveal commonalities or differences in strategy, we analyze and compare the performance of multiple generative behavioral models for humans and monkeys performing a multi-dimensional choice task. In the task implemented, subjects are required to determine by trial and error which of three features of a card is currently rewarded. We find and quantify differences in the strategies of monkeys and humans. Monkeys tend to follow a reinforcement learning strategy while preferring or avoiding stimuli that were previously rewarded/unrewarded, while humans tend to perform a memory-limited ideal inference strategy, where reward is maximized. We conclude that humans and monkeys have distinct but memory-efficient computational frameworks that explain the superior performance of humans compared to monkeys.

4.2 Introduction

Imagine foraging for an unfamiliar berry; one would like to know the features of the berries that correlate with a good taste. Many features could play a part: size, color, softness, access to the sun, etc. Every sample serves as evidence for the positive and negative features that predict taste. However, the number of possible features and feature combinations is challenging to memorize. Through experience, how would one find the relevant features that predict a delicious berry? This question is an example of the credit assignment problem [1], whereby it is difficult to assign credit to the specific features that lead to reward. In these situations, the brain needs to learn through trial and error a subset of specific features that lead to positive, negative, or null outcomes. Investigations of tasks with ambiguous credit assignment problems in a large feature space have the potential to uncover insights into adaptive strategies and computational frameworks necessary to navigate complex environments.

Direct comparisons of behavior across species have shown that monkeys generally perform worse than humans in decision-making and memory-based tasks [54, 86–90, 147], suggesting that the two species may have differing abilities to assign credit, hold features in working memory, and rapidly update memories. Making inter-species comparisons of behavior benefits from the use of computational frameworks that allow one to quantify and compare possible underlying strategies.

To understand species-differentiating decision-making behavior and the contributing computational mechanisms, we need explicit model frameworks. There are many examples that highlight specific strategies. In the ”win-stay lose-shift” (WSLS) model and related variants, subjects focus on a single item, and shift choice after receiving negative feedback; this has been applied to model subject behavior in multi-arm bandit tasks [91, 148–152]. In reinforcement learning strategies [153, 154] and related variants [94, 155], the believed reward value of a choice is updated throughout experience based on

memory-limited feedback. Finally, subjects might employ an ”ideal observer” framework, where logical actions are taken based on information held in memory; generally even ideal strategies should take into account the possibility that memory capacity may be limited [84, 85, 96].

Here, we use these models to evaluate the behavioral data of humans and monkeys performing a multi-dimensional choice task which serves as an exemplar of credit assignment problems with large feature spaces. To compare inter-species differences, the task was designed to be as similar as possible across species. We apply all models and evaluate them on an equal footing; each model probes different aspects of behavior, such as learning rate, memory capacity, and the impact of positive and negative feedback. We then ask: which models fit behavior best across humans and monkeys, and how does the structure and parameters of the best-fitting models inform us about the species-differentiating frameworks used to solve such problems? Fitting these models to data and using them generatively to compare with derived quantities such as the degree of perseveration after a rule switch provides a window into strategy and capability.

We find distinct inter-species strategy differences. Monkeys tend to follow a feature-based reinforcement learning strategy, where they gradually converge to rewarded features and forget non-rewarded features. They also preferentially choose a rewarded card if the same card is presented in the next trial, requiring an amendment of classic reinforcement learning models [94, 155]. Humans, on the other hand, follow an ”ideal observer” like strategy, where they maximally use information to gain reward. By examining choices during critical points in the task, we find that humans are able to perform the correct inference and quickly learn to assign credit to the correct feature. However, humans are unable to remember all features of the task. The major difference between monkeys and humans is the speed with which information is applied and discarded. Humans can quickly forget non-important features while monkeys unnecessarily retain information that distracts their attention. We hypothesize that this difference is in part due to the discrete outcomes of the task, where humans readily understand the concept of features attached to rewards.

In this manuscript, we offer an inter-species generative model comparison of subjects performing credit assignment problems with large feature spaces. We assess models via two metrics, choice-prediction accuracy and reproduction of task behavior, a process necessary for properly describing the computational frameworks of each subject. The differences in the species strategies lead to further understanding of the improved performance of humans over monkeys. The computational frameworks involved may be compared to in the machine learning literature, where research on memory limited algorithms are being introduced to efficiently learn tasks. Lastly, this work is important for understanding the specific computations performed inside the brain. In studies of this task with co-occurring neural recordings, comparing the neural data to our reported computational frameworks can elucidate how the brain flexibly stores, modifies, and accesses critical information upon learning the rewarded features through feedback

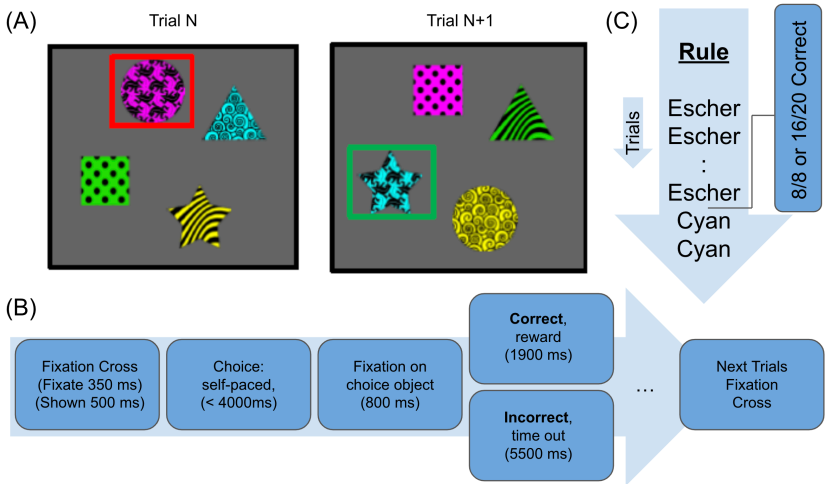

Figure 4.1: Schematic of the multi-dimensional choice task for the monkeys. (A) shows two sample images of what the subjects see on consecutive trials. The subject chooses one of the four displayed objects and either receives a reward or timeout, depending on if they were correct or incorrect, respectively. The green and red boxes indicate sample choices corresponding to a correct and incorrect choice. In this case, the correct choice is the card with the Escher pattern. (B) The timecourse of the task. (C) Underlying rules for consecutive trials. Once the subject chooses 8 out of 8 or 16 out of 20 correct trials in a row, the rule switches. The version performed by humans varied in specifics of timing and trial structure (see main text). The task structure is the same for both species.

and observations.

4.3 Methods

4.3.1 Multi-Dimensional Choice Task

Subjects perform a multi-dimensional choice task (Fig.2.1) in which subjects see four "cards," or images, on a blank screen. Each card has three features: a color, shape, and overlaid pattern. Each of the three features can take one of four values, so that there are 12 unique features. The subject is rewarded if they choose the card with the "correct" feature. The rule is hidden, i.e. never explicitly conveyed to the subject. On every trial, the features are randomly reassigned independently across the four cards. The correct feature, or hidden rule, is held fixed until a criterion is reached. After criterion, the hidden rule is changed to a new, randomly chosen feature in an uncued manner. This task is a variant of the Wisconsin Card Sort Task [95] (WCST), which is used in psychological settings to probe behavioral flexibility. However, the way in which the WCST is typically implemented has significant differences from this framework, which is designed to probe inference of a changing rule.

Trials are initiated for monkeys with a 350 ms fixation onto a cross in the middle of a blank screen. The cross is shown for an additional 150 ms, and then a blank screen appears for 98 ms. The four cards then appear on the screen. Monkeys signal their choice by fixating on a card for 800 ms within

4000 ms of the cards appearing. If the card with the currently rewarded feature is chosen, monkeys are rewarded with a juice/chow mixture and a beeping noise every 200 ms for 1400 ms, after a 98 ms delay. If incorrect, the monkeys receive a time out period of 5100 ms. There is then a 400 ms intertrial interval until the next cross appears. Criterion for monkeys, at which the hidden rule is switched to a new hidden rule, occurs when the subject gets 8 out of 8, or 16 out of 20 trials correct in a row. There is a 50 percent chance of a intra vs extra-dimensional shift, where the correct feature either remains in the same dimension or switches to a new dimension. Every session is initialized with the same rule. set of possible correct features. Sessions end after about 3 hours.

Experiments were repeated in human subjects. For human subjects, trials are initiated by fixating for 300 ms on a cross in the middle of a blank screen. The stimuli appear immediately afterwards. Humans choose a card via an arrow key press within 4000 ms and receive instant visual feedback with a label of "correct" or "incorrect." They then have a 1000 ms intertrial interval. Criterion for humans is 5 out of 5, or 8 out of 10 correct trials in a row. The rule switches completely randomly after criterion is reached. Every session is initialized with the same set of possible correct features. Sessions end after 300 trials for humans.

To remove bias, the first two and last blocks were removed from analysis. Trials with no response were discarded. Some sessions were removed due to bugged implementation. This left us with 29, 29, 86, and 23 sessions for Monkey S, C, B, and T respectively. On average, there were 881 $\pm$ 268 trials per session. There were 5 sessions per human subject, with 267 $\pm$ 9 trials per session.

4.3.2 Computational Models

We summarize the applied computational models. From Niv et al. [94], we adapted a Feature Reinforcement Learning (RL) model, Feature RL with Decay model, Naive model, Serial Hypothesis model, Bayesian Model, and Bayesian Hybrid with Feature RL model. These models were originally designed with different numbers of features and probabilistic rewards. We also added a "smart" serial hypothesis model, an extension of Niv's serial hypothesis model. We adapted a feature based model from Bishara et al. [155], which was originally designed for the Wisconsin Card Sorting Task [95]. We adapted a Win-Stay Lose-Shift model [91]. We also designed two extra models: a Discrete Memory model and a Hybrid Memory model.

One difficulty is the different notation used by Niv [94], Bishara, [155], Nowak [91], and ourselves. To highlight differences between the models, we rewrite every model in the same notation. We define two vectors per trial t: a hidden state vector $\mathbf{v}^{(t)} \in \mathcal{R}^{12}$ and a decision vector $\mathbf{d}^{(t)} \in \mathcal{Z}^{12}$ consisting of 9 zeros and 3 ones. The ones indicate the three chosen features. We define $\mathbf{C}^{(t)} \in \mathcal{Z}^{4\times 12}$, where each row corresponds to a card and each card has 9 zeros and 3 ones. The ones indicate the three features on each card. Additional parameters, hyperparameters, and adjustments are defined in each model. The

framework for every model can be summarized as two general steps: finding the probability of choosing any card in $\mathbf{C}^{(t)}$ based on the hidden state $\mathbf{v}^{(t)}$, and the update to the hidden state $\mathbf{v}^{(t)} \to \mathbf{v}^{(t+1)}$ based on the cards $\mathbf{C}^{(t)}$ and the decision $\mathbf{d}^{(t)}$. We will use lowercase letters to denote rows/columns of matrices, such that $\mathbf{c}_{k:}^{(t)}$ corresponds to the k^{th} card in $\mathbf{C}^{(t)}$.

In many models, we apply a softmax function to find probabilities of choosing each card k:

$$\text{softmax}(\mathbf{x})_k = \frac{e^{\beta x_k}}{\sum_{j=1}^{4} e^{\beta x_j}}, \tag{4.1}$$

where β is a free parameter that controls the bias towards large values. When $\beta = 0$, there is an equal chance of choosing any card. As β grows towards infinity, the probability of choosing the entry with the highest value steadily approaches one, following a *maximum a posteriori* paradigm.

Naive Reinforcement Learning

The Naive RL model [94] is a baseline control model that does not generalize across features. Instead, it uses reinforcement learning to update the belief of individual cards, or collections of three features. This kind of strategy spends an extraordinary amount of time learning the task, and, aside from a purely random strategy, is expected to have a lower bound on performance.

Each entry of the hidden state vector $\mathbf{v}^{(t)} \in \mathcal{R}^{64}$ corresponds to a unique card and is initialized to 0. $\mathbf{d}^{(t)} \in \mathcal{Z}^{64}$ is a one-hot vector, where the one corresponds to the chosen card. Each row of $\mathbf{S}^{(t)} \in \mathcal{Z}^{4\times 64}$ is a one-hot vector, where the one corresponds to a unique card. The probability of choosing a card k is given by the softmax:

$$P_k^{(t)} = \text{softmax}(\mathbf{S}^{(t)}\mathbf{v}^{(t)}), \tag{4.2}$$

The hidden state is updated via a reinforcement learning approach:

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)} + \mathbf{d}^{(t)}\eta(1 - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)}) & \text{if correct} \\ \mathbf{v}^{(t)} + \mathbf{d}^{(t)}\eta(0 - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)}) & \text{if incorrect,} \end{cases} \tag{4.3}$$

where η is the step size of the update.

Win-Stay Lose-Shift

A traditional approach to modeling bandit tasks is the Win-Stay Lose-Shift (WSLS) model [91], where the agent chooses a card based on one feature. If the feature is rewarded, the agent chooses that feature again. Otherwise, the agent chooses one of the other 11 features.

The hidden state vector $\mathbf{v} \in \mathcal{Z}^{12}$ one-hot encodes the hypothesized feature. The probability of

choosing a card is given by the dot product:

$$P_k^{(t)} = \mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)}. \tag{4.4}$$

The agent updates $\mathbf{v}$ from the set of twelve vectors $\mathcal{V} = \{[1, 0, ..., 0], [0, 1, 0, ..., 0], ..., [0, 0, ..., 1]\}$:

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)} & \text{if correct} \\ \text{choose}(\mathcal{V} - \mathbf{v}^{(t)}) & \text{if incorrect} \end{cases}, \tag{4.5}$$

where the hidden state remains the same if correct. If incorrect, one of the vectors in $\mathcal{V}$, not including $\mathbf{v}^{(t)}$, is chosen as the new hidden state. When correct, the hidden state remains the same.

Serial Hypothesis Testing

The Serial Hypothesis model [94] is introduced as a more general WSLS model that allows for random guesses. The one-hot hidden state $\mathbf{v}^{(t)} \in \mathcal{Z}^{12}$ encodes the hypothesized feature. The probability of choosing a card k is

$$P_k^{(t)} = (1 - \epsilon)\mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)} + \frac{(\epsilon)}{3}(1 - \mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)}). \tag{4.6}$$

The card with the feature of interest is chosen $1 - \epsilon$ percent of the time.

Defining a set of twelve vectors $\mathcal{V} = \{[1, 0, ..., 0], [0, 1, 0, ..., 0], ..., [0, 0, ..., 1]\}$, the agent updates $\mathbf{v}$:

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)}(\mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) + \text{choose}(\mathcal{V} - \mathbf{v}^{(t)})(1 - \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) & \text{if correct} \\ \text{choose}(\mathcal{V} - \mathbf{v}^{(t)})(\mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) + \mathbf{v}^{(t)}(1 - \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) & \text{if incorrect.} \end{cases} \tag{4.7}$$

If the feedback is consistent with the hypothesis that the chosen feature returns positive feedback, the hidden state remains the same. Otherwise, the hidden state updates to one of the other 11 possible hidden states.

Smart Serial Hypothesis

In the Serial Hypothesis model, if a card is randomly guessed and correct, the updated hypothesized feature is randomly selected from 11 possible features. A "smart" approach would instead switch the hypothesized feature to one of the three chosen features. To address this possibility, we introduce a "Smart" Serial Hypothesis model. The probability of choosing a card remains the same:

$$P_k^{(t)} = (1 - \epsilon)\mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)} + \frac{(\epsilon)}{3}(1 - \mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)}). \tag{4.8}$$

We define a set of twelve vectors
$\mathcal{V} = \{[1, 0, ..., 0], [0, 1, 0, ..., 0], ..., [0, 0, ..., 1]\}$ and the subset of three vectors $\mathcal{D} \subset \mathcal{V}$ that form the decision $\mathbf{d}$: $\mathcal{D}_i \cdot \mathbf{d} = 1$. The hidden state updates as

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)} & \text{if cor \& } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} = 1 \\ \text{choose}(\mathcal{D}) & \text{if cor \& } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} = 0 \\ \text{choose}(\mathcal{V} - \mathbf{v}^{(t)})(\mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) & \\ \quad +\mathbf{v}^{(t)}(1 - \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}) & \text{if inc,} \end{cases} \tag{4.9}$$

where "cor" and "inc" refer to correct and incorrect. The hidden state now switches to a chosen feature if a card, without the previous hypothesized feature, is correctly chosen.

Bayes Model

Bayes model derives from statistically optimal Bayesian inference [94] to find choices that are most likely to return a reward. The hidden state $\mathbf{v}^{(t)} \in \mathcal{R}^{12}$ captures the probability that each associated feature is the correct feature. Due to their non-deterministic nature of the rewards, Niv et al. [94] defines the probability of receiving a reward given a correct or incorrect choice. We label the probabilities as p and $1 - p$ respectively. The probability of choosing a card k is given by the softmax over $p \times \mathbf{v}$:

$$P_k^{(t)} = \text{softmax}(p\mathbf{C}^{(t)}\mathbf{v}^{(t)}). \tag{4.10}$$

Since the probability of receiving a reward in our task is 100 percent, p does not affect the probability. As the softmax free parameter β grows towards infinity, the probability of a choice becomes deterministic. When $\beta = \inf$, the model is optimal Bayesian.

The vector $\mathbf{v}$ is initialized to $1/12$ for all entries, and it is updated recursively based on the probability of receiving a reward p:

$$\mathbf{v}^{(t+1)} \propto \begin{cases} p\mathbf{d}^{(t)} \odot \mathbf{v}^{(t)} + (1-p)(\mathbf{1} - \mathbf{d}^{(t)}) \odot \mathbf{v}^{(t)} & \text{if correct} \\ (1-p)\mathbf{d}^{(t)} \odot \mathbf{v}^{(t)} + p(\mathbf{1} - \mathbf{d}^{(t)}) \odot \mathbf{v}^{(t)} & \text{if incorrect.} \end{cases} \tag{4.11}$$

Since in our task $p = 1$, the above equation is simplified to

$$\mathbf{v}^{(t+1)} = \begin{cases} \frac{\mathbf{d}^{(t)} \odot \mathbf{v}^{(t)}}{\mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)}} & \text{if correct} \\ \frac{(\mathbf{1} - \mathbf{d}^{(t)}) \odot \mathbf{v}^{(t)}}{(\mathbf{1} - \mathbf{d}^{(t)}) \cdot \mathbf{v}^{(t)}} & \text{if incorrect.} \end{cases} \tag{4.12}$$

The update rule in Niv et al. [94] only works when the rule does NOT change. In our version, where

the rule is able to switch, we modify the update equation: when the dot product is 0, we re-initialize $\mathbf{v}$ to 1/12 and re-evaluate the update equation.

Discrete Memory Model

We introduce the Discrete Memory Model to model memory limited "optimal" behavior, where it maximizes the chance of getting reward on every trial based on the remembered positively or negatively rewarded features. We define the vector $\mathbf{v} \in \mathcal{Z}^{12}$ as a vector of zeros, ones, and negative ones. Features with $v_i = 1$ are preferentially chosen. Features with $v_i = -1$ are preferentially avoided. Features with $v_i = 0$ are chosen at the chance level. To enforce the memory limitations, there are at most N_C positive ones and N_I negative ones in $\mathbf{v}$. Any entry equal to 0 is considered to be a feature with unknown information. For statistically optimal behavior when $N_C = 3$, $\mathbf{v}^{(t)}$ contains either ones and zeros OR negative ones and zeros: once correct, remembering negative information is unhelpful. For example, if card A has two positively rewarded features and one negatively rewarded feature and card B has one positively rewarded feature and two unknown features, card A should be chosen due to the greater amount of positively rewarded features. Negatively rewarded features do not give more information than information gained from positively rewarded features. When $N_C < 3$ and $N_I > 0$, there are edge cases when it's best to include both positively and negatively rewarded features. We ignore these cases and force $\mathbf{v}$ to include either positive or negative ones for each trial.

The subject may make a choice before or after seeing the cards. If choosing before, the subject can only use information from previous trials: they would randomly choose a feature that was positively rewarded or a feature that was not negatively rewarded. If choosing after, the subject can choose a card with the largest number of positively rewarded features or least number of negatively rewarded features. We can write this with a continuous parameter β as:

$$P_k^{(t)} = \begin{cases} \epsilon\frac{1}{4} + (1-\epsilon)\frac{(\mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)})^\beta}{\sum_{j=1}^4 (\mathbf{c}_{j:}^{(t)} \cdot \mathbf{v}^{(t)})^\beta} & \text{if } \sum_i v_i > 0 \\ \epsilon\frac{1}{4} + (1-\epsilon)\frac{(3+\mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)})^\beta}{\sum_{j=1}^4 (3+\mathbf{c}_{j:}^{(t)} \cdot \mathbf{v}^{(t)})^\beta} & \text{if } \sum_i v_i < 0 \\ \frac{1}{4} & \text{if } \sum_i v_i = 0. \end{cases} \tag{4.13}$$

As β increases, there is a higher likelihood of choosing a card with the largest sum. There is ϵ noise dictating the choice.

The hidden state $\mathbf{v}^{(t)}$ is updated based on "optimal" logical reasoning. We define "rewarded features" as features that previous gave a positive reward or $v_i = 1$. We define non-rewarded features as features that previous gave a negative reward or $v_i = -1$. We define forgetting as setting v_i to 0. The logical reasoning is the following: if correct and the chosen card has 1-3 rewarded features, forget about any non-chosen feature. If correct and the chosen card has 0-2 non-rewarded features, remember chosen

features that were not previously non-rewarded. If incorrect, choosing a card with rewarded features, and there exist non-chosen rewarded features, remember the non-chosen rewarded features. If incorrect, choosing a card with rewarded features, and there does not exist non-chosen rewarded features, forget everything and remember that the chosen features are all non-rewarded; this typically occurs when the rule changes. Lastly, if incorrect, 0-3 non-rewarded chosen features, and the non-chosen features are all also either unknown or non-rewarded, then add the chosen features to the list of non-rewarded features. If incorrect, 0-3 non-rewarded chosen features, and there exists non-chosen rewarded features, then do nothing to the remembered features. The logical reasoning is mathematically written as

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)} & \text{if cor } \& \ \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} > 0 \\ \mathbf{d}^{(t)} + \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)} & \text{if cor } \& \ \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} \leq 0 \\ \mathbf{v}^{(t)} - \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)} & \text{if inc, } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} > 0 \\ & \quad \& \ \sum(\mathbf{v}^{(t)} - \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)}) > 0 \\ -\mathbf{d}^{(t)} & \text{if inc, } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} > 0 \\ & \quad \& \ \sum(\mathbf{v}^{(t)} - \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)}) = 0 \\ \mathbf{v}^{(t)} - (\mathbf{1} + \mathbf{v}^{(t)}) \odot \mathbf{d}^{(t)} & \text{if inc, } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} \leq 0 \\ & \quad \& \ \sum(\mathbf{v}^{(t)} - \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)}) \leq 0 \\ \mathbf{v}^{(t)} & \text{if inc, } \mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)} \leq 0 \\ & \quad \& \ \sum(\mathbf{v}^{(t)} - \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)}) > 0 \end{cases}, \tag{4.14}$$

where ”cor” and ”inc” refer to correct and incorrect. The memory limitation is added after the logic of computing $\mathbf{v}^{(t+1)}$: if the number of positive (negative) ones is greater than $N_C(N_I)$, the model randomly selects $N_C(N_I)$ positive (negative) ones and sets the rest to 0.

Hybrid Memory Model

We introduce the Hybrid Memory model as a multi-feature WSLS model, as postulated in Goudar et al. [147]. Instead of holding multiple features in memory with equal probabilities, one feature is focused on with win-stay lose-shift behavior while ”backup” features are remembered in case of incorrect feedback. At any point, the hypothesized feature or backup features may be focused on. We define a one-hot ”hypothesis” vector $\mathbf{h} \in \mathcal{Z}^{12}$ and a ”memory” vector $\mathbf{m} \in \mathcal{Z}^{12}$. The memory vector $\mathbf{m}$ contains $2 \leq N_C \leq 3$ ones and $12 - N_C$ zeros. Ones indicate the features held in memory. The hypothesis is a subset of the memory: $\mathbf{h} \cdot \mathbf{m} = 1$.

For any trial t, the probability of choosing a feature $\mathbf{v}^{(t)}$ is calculated as:

$$\mathbf{v}^{(t)} = \begin{cases} \alpha\mathbf{h}^{(t)} + \frac{(1-\alpha)\beta}{N_C-1}(\mathbf{m}^{(t)} - \mathbf{h}^{(t)}) + \frac{(1-\alpha)(1-\beta)}{12-N_C}(\overline{1} - \mathbf{m}^{(t)}) & \text{if correct} \\ \gamma\mathbf{h}^{(t)} + \frac{(1-\gamma)\nu}{N_C-1}(\mathbf{m}^{(t)} - \mathbf{h}^{(t)}) + \frac{(1-\gamma)(1-\nu)}{12-N_C}(\overline{1} - \mathbf{m}^{(t)}) & \text{if incorrect} \end{cases} \tag{4.15}$$

The first term is the probability of choosing the hypothesis. The second term is the probability of choosing a feature in memory that isn't the hypothesis. The third term is the probability of choosing a feature not in memory. The probability of choosing any card is

$$P_k^{(t)} = \mathbf{c}_{k:}^{(t)} \cdot \mathbf{v}^{(t)}. \tag{4.16}$$

A single feature from the chosen card is selected to be the new hypothesis $\mathbf{h}^{(t+1)}$ with probability given by $\mathbf{v}^{(t)}$:

$$\frac{\mathbf{v}^{(t)} \odot \mathbf{d}^{(t)}}{\mathbf{v}^{(t)} \cdot \mathbf{d}^{(t)}}. \tag{4.17}$$

If correct, the memory is updated from the previous guess to include the previous decision and new hypothsis:

$$\mathbf{m}^{(t+1)} = \mathbf{h}^{(t+1)} + \text{choose}(N_C - 1)[\mathbf{d}^{(t)} - \mathbf{h}^{(t+1)}] \tag{4.18}$$

"choose" refers to choosing $N_C - 1$ samples without replacement from $\mathbf{d}^{(t)} - \mathbf{h}^{(t+1)}$. If incorrect, the memory is updated to include the new hypothesis $\mathbf{h}^{(t+1)}$ and other features of differing dimension:

$$\mathbf{m}^{(t+1)} = \mathbf{h}^{(t+1)} + \text{choose}(N_C - 1)[\mathbf{m}^{(t)}[\mathbf{m}_{dim}^{(t)} != \mathbf{h}_{dim}^{(t+1)}]]. \tag{4.19}$$

This model has elements of the WSLS model and the Discrete Memory model. If correct, an ideal agent would continue to choose the hypothesis ($\alpha = 1$). If incorrect, the agent would test other things in memory or switch to a random guess ($\gamma = 0$, $\nu \leq 1$). The main difference occurs when ν is between 0 and 1: features are remembered from the previous correct trial and random features can be chosen.

Feature Reinforcement Learning

The Feature Reinforcement Learning (RL) model [94] is implemented to assess the ability to gradually learn information about individual features. The hidden state $\mathbf{v} \in \mathcal{R}^{12}$ is initialized to zero. The probability of choosing a card is given by:

$$P_k^{(t)} = \text{softmax}(\mathbf{C}^{(t)}\mathbf{v}^{(t)}). \tag{4.20}$$

The vector $\mathbf{v}$ is updated towards the value r with a step-size of η depending on the result of the trial:

$$\mathbf{v}^{(t+1)} = \mathbf{v}^{(t)} + \eta(r - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)})\mathbf{d}^{(t)} \tag{4.21}$$

where $r = 1$ if the trial was correct and $r = 0$ if the trial was incorrect.

Feature Reinforcement Learning with Decay

To add memory limitations, the Feature RL with Decay model [94] expands on the Feature RL model by gradually "forgetting" non-chosen features. The probability of choosing a card is given by:

$$P_k^{(t)} = \text{softmax}(\mathbf{C}^{(t)}\mathbf{v}^{(t)}). \tag{4.22}$$

The hidden state $\mathbf{v}$ is updated with:

$$\mathbf{v}^{(t+1)} = (1 - \gamma)\mathbf{v}^{(t)} \odot (\mathbf{1} - \mathbf{d}^{(t)}) + \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)} + \eta(r - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)})\mathbf{d}^{(t)}, \tag{4.23}$$

where $r = 1$ if correct and $r = 0$ if incorrect. Now, non-chosen features decay towards zero when $\gamma > 0$.

Extended Feature Reinforcement Learning

Motivated by behavior during critical regions of the task, we modify the Feature RL with Decay model into the Extended Feature RL model. First, it's plausible that there are different step-size and decay rates for correct and incorrect features. In the Feature RL model, the step-size is used to update each hidden value towards the arbitrary reward value of 1 or 0 depending on the feedback. Changing the reward value necessarily changes the best fit step-size. Since both reward values were arbitrarily chosen, one cannot guarantee that the step-size is equal valued for both correct and incorrect trials. We argue that either independent step-sizes or fitted reward values are necessary for the best fit models of this task. We allow for independent step-sizes and maintain the constant reward values. The decay rates should also be independent between correct and incorrect feedback, since there is an interplay between step-size and decay rates.

Second, we're motivated by Fig. 4.6, where previously chosen cards are chosen/avoided with high likelihood in monkeys. Thus, we add a boost or decay to each softmax term if the card is the exact same, or extremely similar, as the previously chosen card. Specifically, α_{3C} and α_{3I} are added/subtracted if the previously chosen correct/incorrect card and card k are the same. Similarly, if the previously chosen correct/incorrect card and card k have two of the same features, α_{2C} and α_{2I} are added/subtracted. To simplify, we use α_{ig} to refer to each of the previously defined values depending on if the previous trial

was correct ($g = C$) or incorrect ($g = I$). The probabilities of choosing each card become.

$$P_k^{(t)} = \text{softmax}(\mathbf{C}^{(t)}\mathbf{v}^{(t)} + \sum_{i=2}^{3} \alpha_{ig}\delta(\mathbf{C}^{(t)}\mathbf{d}^{(t-1)} - i\mathbf{1})) \tag{4.24}$$

The update of $\mathbf{v}$ is modified by allowing different step-sizes and decay rates for correct and incorrect trials. Further, the additional α's are added based on the previous and current feedback to account for the adjustments in choice probabilities. To simplify, we use η_f and γ_f to refer to the step-sizes and decay rates for correct trials $f = C$ and incorrect trials $f = I$. The hidden values update as:

$$\begin{aligned}\mathbf{v}^{(t+1)} &= (1-\gamma_f)\mathbf{v}^{(t)} \odot (\mathbf{1} - \mathbf{d}^{(t)}) + \mathbf{v}^{(t)} \odot \mathbf{d}^{(t)} \\ &\quad + \eta_f \left(r - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)} - \sum_{i=2}^{3} \alpha_{ig}\delta(\mathbf{d}^{(t)} \cdot \mathbf{d}^{(t-1)} - i) \right) \mathbf{d}^{(t)},\end{aligned} \tag{4.25}$$

where $r = 1$ if correct and $r = 0$ if incorrect. Note that $\mathbf{v}$ updates differently depending both the current feedback and previous feedback, leading to four different cases.

Bayesian Hybrid with Feature RL

The Bayesian Hybrid model [94] was introduced as a way to combine reinforcement learning with a focus on particular dimensions. Niv et al. hypothesized that the subjects may Bayesian optimally learn the feature dimension, which may impact the choice of each card. They use a vector $\mathbf{v} \in \mathcal{R}^{12}$ and a dimensional attention vector $\phi \in \mathcal{R}^{12}$. To aid in writing, we define the matrix $\mathbf{G} \in \mathcal{Z}^{3\times 12}$, where each row has 4 ones corresponding to the four features in it's respective dimension. The probability of choosing each card is:

$$P_k^{(t)} = \text{softmax}(\mathbf{C}^{(t)} \cdot (\phi^{(t)} \odot \mathbf{v}^{(t)})). \tag{4.26}$$

To update the dimensional attention, the model first finds the probability each feature is rewarding:

$$\mathbf{p}^{(t+1)} = \begin{cases} \frac{\mathbf{d}^{(t)} \odot \mathbf{p}^{(t)}}{\mathbf{d}^{(t)} \cdot \mathbf{p}^{(t)}} & \text{if correct} \\ \frac{(\mathbf{1}-\mathbf{d}^{(t)}) \odot \mathbf{p}^{(t)}}{(\mathbf{1}-\mathbf{d}^{(t)}) \cdot \mathbf{p}^{(t)}} & \text{if incorrect.} \end{cases} \tag{4.27}$$

The vector $\mathbf{p}$ updates in the same way as the Bayesian model. The probabilities are summed across a dimension k to get dimension weights:

$$\phi^{(t+1)} = \sum_{k=1}^{3} \mathbf{g}_k \frac{(\mathbf{g}_k \cdot p^{(t+1)})^\alpha}{\sum_{j=1}^{3} (\mathbf{g}_j \cdot p^{(t+1)})^\alpha}. \tag{4.28}$$

The hidden state $\mathbf{v}$ is updated with reinforcement learning, weighted by dimensional attention:

$$\mathbf{v}^{(t+1)} = \begin{cases} \mathbf{v}^{(t)} + \eta(1 - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)})\mathbf{d}^{(t)} \odot \phi^{(t+1)} & \text{if correct} \\ \mathbf{v}^{(t)} + \eta(0 - \mathbf{d}^{(t)} \cdot \mathbf{v}^{(t)})\mathbf{d}^{(t)} \odot \phi^{(t+1)} & \text{if incorrect.} \end{cases} \tag{4.29}$$

Sequential Learning Model

Lastly, we adapt the Sequential Learning model introduced in Bishara et al. [155]. This model allows for unique updates to all individual features. This highlights a major difference from the reinforcement learning models, where all chosen features are updated in the same way. They start with a normalized attentional weight vector, which we refer to as the hidden state $\mathbf{v} \in \mathcal{R}^{12}$ for consistency in this manuscript.

The probability of choosing a card k is

$$P_k^{(t)} = \frac{\mathbf{c}_{k:}^{(t)} \cdot (\mathbf{v}^{(t)})^d}{\sum_{j=1}^{4} \mathbf{c}_{j:}^{(t)} \cdot (\mathbf{v}^{(t)})^d} = \frac{\mathbf{c}_{k:}^{(t)} \cdot (\mathbf{v}^{(t)})^d}{\sum_{j=1}^{12} (v_{j:}^{(t)})^d}, \tag{4.30}$$

where d is referred to as the decision-consistency parameter. If $d = 0$, the probability of choosing any card goes to chance. If $d = 1$, the probability sums across the attention of each feature on each card. As d increases above 1, the probabilities become deterministic.

In Bishara et al. [155], an intermediate vector $\mathbf{s}$ is used to update $\mathbf{v}$. We omit this intermediate step and directly write the update to $\mathbf{v}$:

$$\mathbf{v}^{(t+1)} = \begin{cases} (1 - r)\mathbf{v}^{(t)} + r\frac{\mathbf{d}^{(t)} \odot (\mathbf{v}^{(t)})^f}{\sum_{j=1}^{12} d_j^{(t)} (v_j^{(t)})^f} & \text{if correct} \\ (1 - p)\mathbf{v}^{(t)} + p\frac{(1-\mathbf{d}^{(t)}) \odot (\mathbf{v}^{(t)})^f}{\sum_{j=1}^{12} (1-d_j^{(t)}) (v_j^{(t)})^f} & \text{if incorrect} \end{cases}. \tag{4.31}$$

The parameters $0 \leq r \leq 1$ and $0 \leq p \leq 1$ indicate how much the hidden state $\mathbf{v}$ is maintained compared to updated based on choice. The parameter f refers to how v updates from the decision. If $f = 0$, attention is not considered. If $f = 1$, the current attention is weighted. As f approaches infinity, the signal vector approaches zeros and ones.

4.3.3 Model Fitting

To find hyperparameters for each model, we follow the forward modeling [156] approach in Niv [94]. In this approach, the likelihood of the model and subject selecting the same card on trial t is calculated based on the subjects trial history before trial t. Then, we find the hyperparameters that minimize the negative log-likelihood of the model and subject selecting the same card on all trials. This method allows for the computation of hyperparameters that leads to the best one-step prediction accuracy. Then, we can compute the hidden state gathered at every trial based on the subjects trial history and the

hyperparameters.

There are two sets of models that require marginally different approaches when fitting. The first set include the Feature RL, Feature RL with Decay, Extended Feature RL, Bayes, Sequenctial Learning Model, Naive, and Bayesian Hybrid model. In this set of models, the hidden state update is deterministic. That is, the entries in the hidden state $\mathbf{v}^{(t)}$ update based on the previous feedback, choice, cards seen, and hyperparameters. The probability $P_k^{(t)}$ of choosing a card $\mathbf{c}_{k:}^{(t)}$, the card chosen by the subject on trial t, can be expanded generally as:

$$P_k^{(t)} = p(\mathbf{c}_{k:}^{(t)} = \mathbf{d}_s^{(t)} | \mathbf{\Theta}_m, \mathbf{v}^{(t)}), \tag{4.32}$$

where $\mathbf{\Theta}_m$ is a placeholder for model m's hyperparameters and s refers to the subject. To fit these models, the hidden state is updated based on the subjects decision and observations:

$$\mathbf{v}^{(t+1)} = f_m(\mathbf{v}^{(t)}, \mathbf{\Theta}_m, \mathbf{d}_s^{(t)}, \mathbf{C}_s^{(t)}, \text{feedback}_s), \tag{4.33}$$

where f_m is a model specific update function. The entries in the hidden state $\mathbf{v}^{(1)}$ are initialized to 0 or normalized to 1/12 depending on the model. The negative log-likelihood of the model choosing the same card as the subject for every trial can then be calculated as:

$$-\log_2(L) = -\sum_t \log_2(P_k^{(t)}). \tag{4.34}$$

The hyperparameters $\mathbf{\Theta}_m$ that minimize the negative log-likelihood are found with a non-gradient based approach (Nelder-Mead [157]). Base 2 is used such that the average chance performance (25 percent) is equivalent to a value of 2.

The second set of models includes the Discrete Memory, WSLS, Serial Hypothesis, Smart Serial Hypothesis, and Hybrid Memory model. In this set of models, the hidden state updates in a probabilistic way. There are a discrete set of hidden states $\mathbf{v}_{\text{hs}}$. Every trial, the hidden state transitions to any of the possible hidden states with a discrete probability distribution based on the previous feedback, choice, cards seen, and hyperparameters. To fit these models, one needs to marginalize across all possible hidden states (hs). The probability $P_k^{(t)}$ of choosing a card $\mathbf{c}_{k:}^{(t)}$, the card chosen by the subject on trial t, is expanded as:

$$P_k^{(t)} = \sum_{\text{hs}} p(\mathbf{c}_{k:}^{(t)} = \mathbf{d}_s^{(t)} | \mathbf{\Theta}_m, \mathbf{v}_{\text{hs}}) p^{(t)}(\mathbf{v}_{\text{hs}}). \tag{4.35}$$

The probability of being in any hidden state is updated based on the subjects decision and observations:

$$p^{(t+1)}(\mathbf{v}_{(\text{hs\_new})}) = \sum_{\text{hs}} p(\mathbf{v}_{(\text{hs\_new})} | \mathbf{v}_{(\text{hs})}, \mathbf{\Theta}_m, \mathbf{d}_s^{(t)}, \mathbf{C}_s^{(t)}, \text{feedback}_s) p^{(t)}(\mathbf{v}_{(\text{hs})}). \tag{4.36}$$

The probabilities of each hidden state are normalized based on the number of hidden states. The negative log-likelihood of the model choosing the same card as the subject for every trial is again with Eq. 4.34. The hyperparameters $\mathbf{\Theta}_m$ that minimize the negative log-likelihood are again found with a non-gradient based approach (Nelder-Mead [157]). For the Discrete Memory Model, the negative log-likelihood is found independently for all combinations of N_C and N_I, and only the best combinations are reported.

4.3.4 Behavioral Statistics Comparison

To validate the models, we compare subjects behavioral statistics with the statistics generated from each model. Statistics of interest are length of perseveration after a rule switch, speed of learning from correct feedback, and integration of choices and feedback from the previous trial into the current trial. These statistics are computed and averaged across all sessions for each subject.

Since these are generative models, once the hyperparameters are found, the models can be ran on simulated sessions. We simulate sessions in the exact same way as performed by each subject. For the independent hyperparameters found for every session, the model is ran on 100 times the number of trials completed by the subject. Model statistics are then computed and averaged across sessions.

4.4 Results

To understand the inter-species differences in computational frameworks when solving ambiguous credit assignment problems with large feature spaces, we train and test non-human primates and humans on a multi-dimensional choice task. In this task, the subject needs to choose one of four cards, each with a unique color, shape, and pattern. Only one of the twelve features rewards the subject. We begin by fitting a plethora of behavioral models, including win-stay lose-shift like models, naive models, feature-based models, and memory-limited ideal-observer models. Models are abbreviated according to Tab. 4.1. Per subject, models are evaluated with the negative log likelihood of generating the same sequence of card selections as the subject. To complement negative log-likelihood based model evaluation, we additionally define a few critical behavioral statistics of the task. We validate models on their ability to reproduce such statistics per subject. To help explain behavioral statistics, we find the need for a feature-based model with flexible choice behavior based on the previous selected card. Model structure, along with behavioral statistics, reveal stark differences in the computational frameworks of humans compared to monkeys.

Abbreviation	Model Name
FRL	Feature Reinforcement Learning
FD	FRL with Decay
FE	Extended FRL
DM	Discrete Memory Model
Ba	Bayes
SL	Sequential Learning Model
N	Naive
WSLS	Win-Stay Lose-Shift
SH	Serial Hypothesis
SSH	Smart Serial Hypothesis
Hy	Bayesian Hybrid Model
HyM	Hybrid Memory Model

Table 4.1: Abbreviations of each model name.

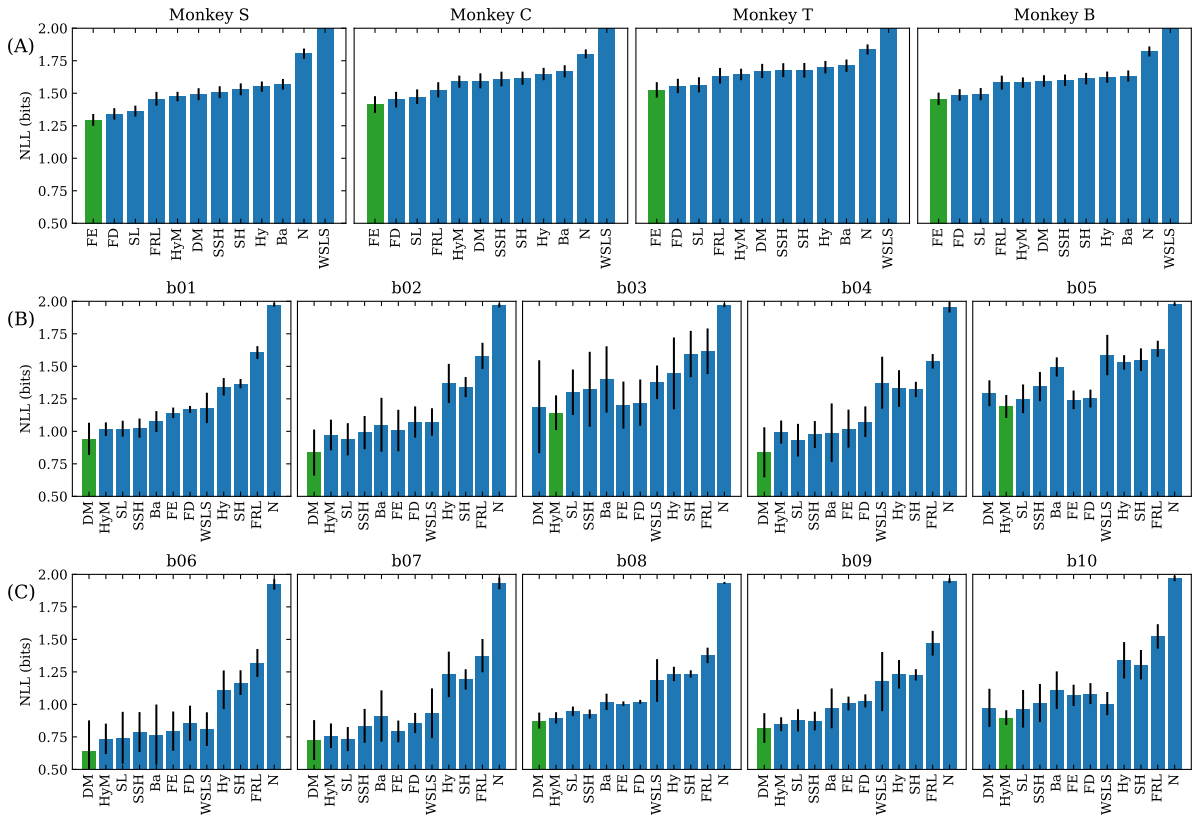

Figure 4.2: Average negative log-likelihood (NLL) for models fit to non-human primates (row A), humans with instruction (row B), and humans with minimal instruction (row C). NLL is calculated with Eq. 4.34, divided by the number of trials. Models are sorted by Monkey S in (A), subject b01 in (B), and subject b06 in (C). Abbreviations are labeled in Tab. 4.1. The best fit models are in green. We consistently see that the best fitting models for monkeys is the extended feature reinforcement learning models, and the best fitting models for humans is the discrete memory models with 2 to 3 correct and incorrect objects in memory or the hybrid memory model. We see the same results with AIC and BIC (4.9 and 4.10), with the exception of Monkeys B., C., and T., where the BIC performance is approximately the same between the extended feature RL model and the feature RL model with decay. The baseline is a random agent: with a 25 percent likelihood of choosing the correct card, the NLL would be 2. With the exception of the WSLS model in the monkeys, all models are better than chance. The bad performance of the WSLS in monkeys is due to the lack of noise and random choices in the model.

4.4.1 Best Fitting Models for Monkeys vs Humans

The model performance for all subjects is shown in Fig. 4.2. The best fitting model for monkeys is the extended feature RL model (Sec. 4.3.2), followed by the feature RL with decay (Sec. 4.3.2). These two models imply that monkeys typically follow a reinforcement learning strategy with a memory limitation. This strategy tracks the hidden values of potential rule features, and values are gradually updated via a temporal difference error. Non-chosen features gradually decay towards zero. The increased performance of the extended feature RL model occurs based on two sets of additional parameters. First, the extended version includes a increase/decrease in probability of choosing a card that matches the previously chosen card. This implies a possible difference in frameworks when a card is repeated compared to a complete mixture of 12 features. Second, the extended version includes differences in step and decay rate depending on feedback, which imply that monkeys may update their beliefs differently depending on trial history.

To account for the differences in number of parameters between models, we also use the Akaike information criterion (AIC) and Bayesian information criterion (BIC) to evaluate model performance. We find consistent results with AIC (Fig. 4.9). We find that feature RL with decay and the extended feature RL have similar BIC (Fig. 4.10) for three out of four monkeys. The extended feature RL model performs better for Monkey S There is a trade off between negative log-likelihood performance and fitting with the minimal number of parameters. Given the similar BIC performance, we focus on the extended feature RL.

For humans, the best fitting models tend to either be the discrete memory model (DM) or the hybrid memory model (HyM) (Fig. 4.2). The discrete memory model is an ideal-observer model with limited memory. The hybrid memory model is a multiple feature win-stay lose-shift model. This model remembers previously correct features and switch between features if the chosen one was incorrect. This strategy can be characteristic of an ideal-observer model, since an ideal observer would continually test previously chosen correct features. We find the same result with AIC and BIC (4.9, 4.10).

There is larger inter-subject variability than the difference between the best fitting models for humans with instruction (Fig. 4.2(B)) compared to humans with minimal instruction (Fig. 4.2(C)). In both cases, the best fitting models are either the discrete memory model or the hybrid memory model. These models suggest that humans employ a memory-limited ideal-observer strategy. In both cases, the strategy keeps track of a few candidate features and switches between these features upon feedback.

The structural differences in the best fit models (Fig. 4.2) point towards two distinct frameworks for monkeys and humans: slow reinforcement learning for monkeys and fast ideal observer behavior for humans.

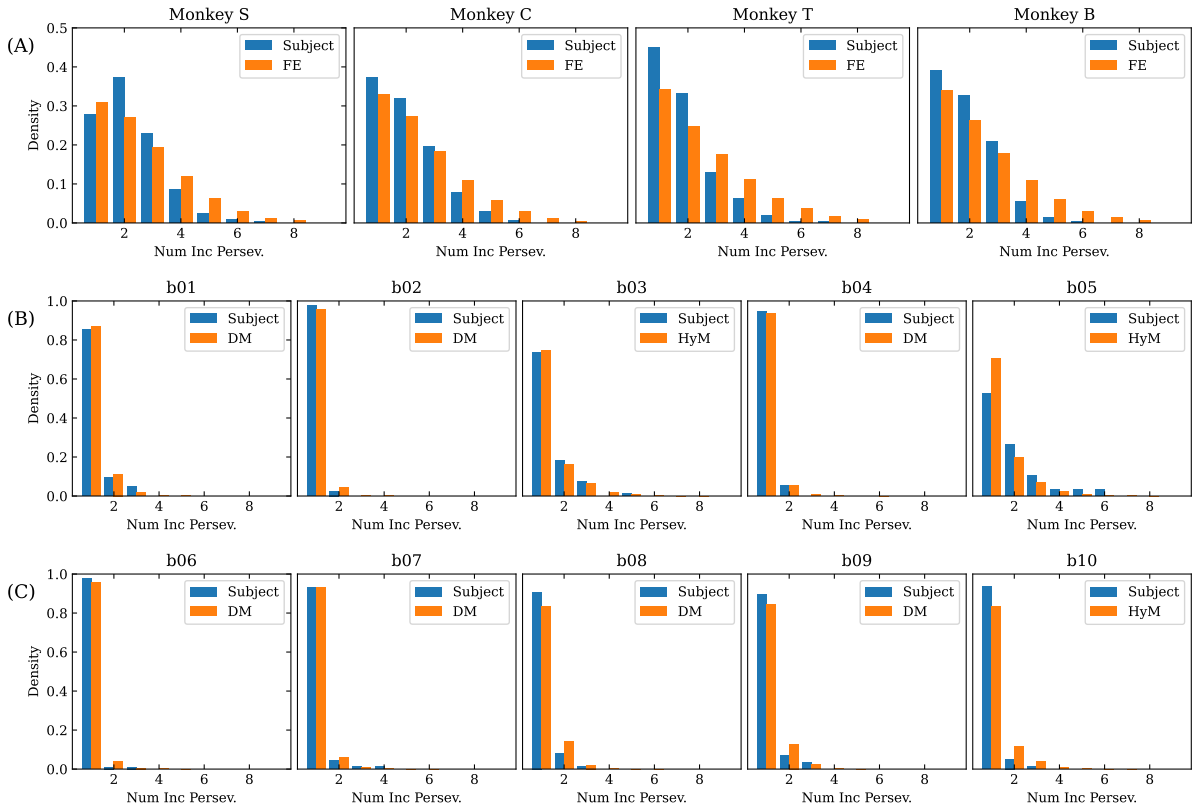

Figure 4.3: The perseveration, or choice of the previous rule feature, of non-human primates (row A), humans with instruction (row B), and humans with minimal instruction (row C). After a rule switch, given a minimum of one incorrect perseveration, the x-axis is the number of incorrect perseverations that occurred before choosing a card without the previous rule feature on it. Monkeys perseverate for a few trials. Humans typically either avoid the previous rule feature or switch to a random choice after receiving incorrect feedback. Each best fit model is simulated on the task, and the statistics match the subjects behavior.

4.4.2 Behavioral Statistics Support Reinforcement Best Fit Models

To validate the identified computational frameworks, we run the best fit models on simulations of the task. Comparisons of behavioral statistics between subjects and models provide further evidence of the models ability to capture subjects strategies.

Evaluation of Rule Switches

When the rule switches and when the subject first receives negative feedback, there are three possible behaviors. First, ideal behavior would begin only choosing cards without the previous rule feature due to knowledge that the rule must have switched. A second behavior, where the subject begins to choose randomly, can occur if the subject is knowledgeable of the rule switching but doesn't follow ideal behavior. If the subject "resets" all internal beliefs, they may begin choosing randomly without any preference towards or away from the previous rule. The third behavior, which we refer to as continuous perseveration, occurs when the subject continually chooses cards with the previous rule feature. Learning is slower, and negative feedback is not quickly attributed to the previous rule feature.

To compare the three possible behaviors, we observe the post rule switch statistics of individual trial perseveration, or choosing a card with the previous rule feature. Specifically, we check the number of times the subjects or best fitting models perseverate and receive incorrect feedback before selecting a card without the previous rule feature (Fig. 4.3). In ideal behavior, all cases would have a single incorrect perseveration. If the subject switched to random behavior, they would perseverate 25 percent of the time. The density in Fig. 4.3 would decay geometrically, following $(1/4)^N$, where N is the number of incorrect perseverations. Lastly, if the subject continuously perseverated, they would have a high number of incorrect perseverations. In Fig. 4.3, the density would decay slower than $(1/4)^N$.

The post rule-switch behavior shows drastic inter-species differences. Monkeys tend to perseverate for an extended amount of time (Fig. 4.3(A), blue bars), which aligns with a slow reinforcement learning strategy (Fig. 4.3(A), orange bars). A slow RL model would continually perseverate until the hidden values associated with the previous rule decays enough that other features are more likely to be selected. Humans, on the other hand, tend to perform ideal behavior and avoid the previous rule (Fig. 4.3(B, C), blue bars). With the exception of subject b03 and b05, the subjects mainly avoid the previous rule feature after a single incorrect perseveration. Subjects b03 and b05 seem to either minorly perseverate or switch to a random behavior. Further, the statistics of all best fit models closely match the subjects behavior (Fig. 4.3(B, C), orange bars). The best fitting models for monkeys are reinforcement learning models, and the best fitting models for humans have ideal observer like behavior.

In the supplemental material, we explore the possibility of different statistics depending on the rule switch type. I.e., when the rule switches dimensions or remains in the same dimension. We find no significant differences in the perseveration statistics (Fig. 4.11) depending on rule switch type.

Speed of Learning the Rule

Different computational frameworks can lead to different speeds of learning from correct trials. Imagine a scenario when the subject is correct twice in a row, and only the rule feature appears on both cards (Fig. 4.4(A)). An ideal observer would immediately learn the rule, and they would choose the correct feature until the rule switches. A subject that randomly selects cards would choose the rule feature 25 percent of the time. It's possible for a non-ideal observer to choose the rule feature more often than a randomly selecting observer and less than an ideal observer. A non-ideal observer could take multiple trials to correctly attribute feedback to the rule feature.

To analyze the speed of learning from correct feedback, we focus on the sequence of $N \geq 2$ correct trials after an initial incorrect trial, used to identify the start of learning. We reject sequences of trials where all N chosen cards share two or three features. E.g., we do not include the case when a cyan triangle is chosen twice in a row since an ideal observer cannot figure out the rule from these two trials. After N trials, we check the percentage that the subject chooses the card containing the rule feature.

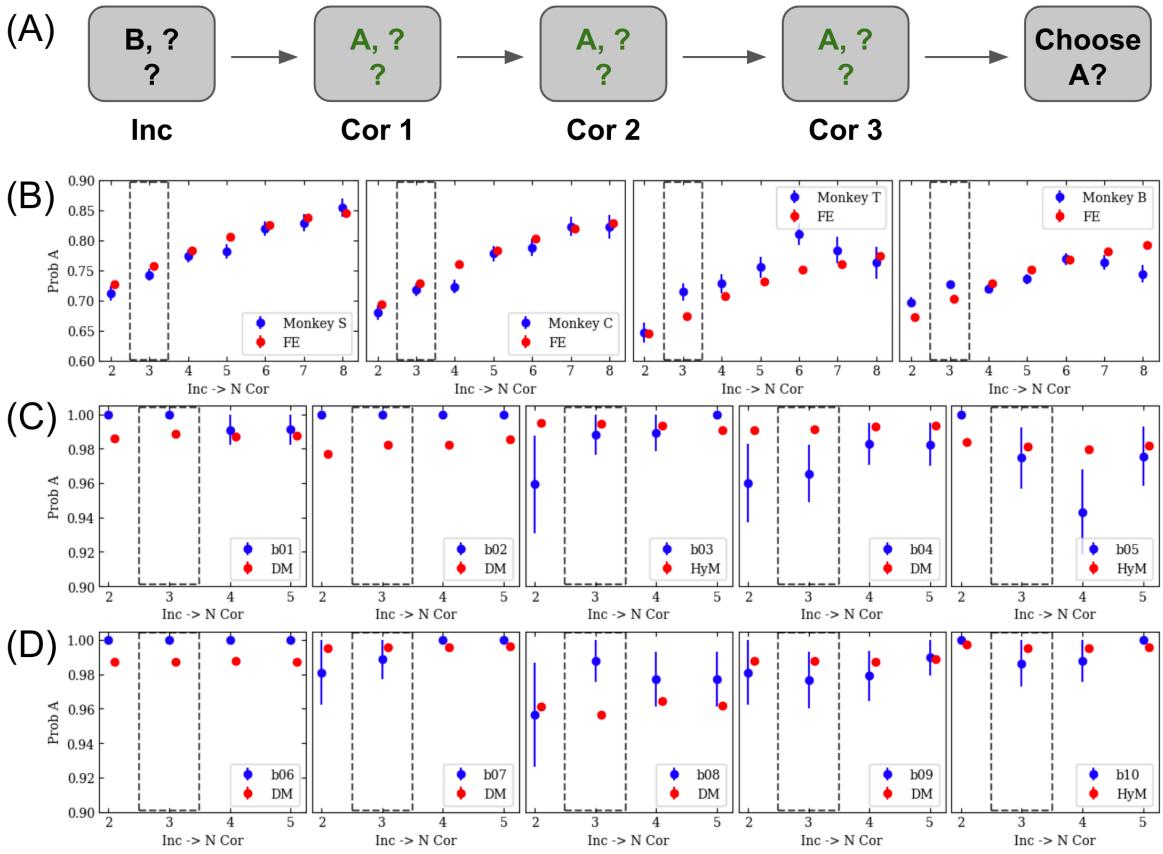

Figure 4.4: Example schematic of behavioral statistic (A). After incorrectly selecting a card, feature A is correctly chosen 3 times in a row. The question is how likely the subject then chooses feature A on the next trial. This example corresponds to the dotted boxes in (B-D). The likelihood of choosing the rule feature if the rule feature is the only consistently chosen feature among $N \geq 2$ correct trials in a row for monkeys (B), humans with instruction (C), and humans with minimal instruction (D). The x-axis labels N. The count is initialized with an incorrect trial before the N correct trials. A subject choosing randomly would choose the rule feature 25 percent of the time. Monkeys tend to increase the percentage as a function of the number of correct trials, consistent with a reinforcement learning architecture. Humans tend to stagnate at or close to 100 percent, consistent ideal strategy architectures. The red dots show the best fit model behavior when simulated. The behavior of the agents tend to match that of the subjects, with the exception of human subjects b01, b02, b06, and b07, where the agent's percentage is lower than the subjects behavior.

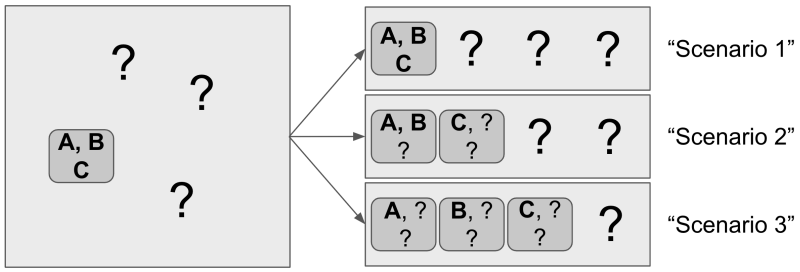

Figure 4.5: Schematic of feature splitting. On the first trial (left), features A, B, and C are all chosen. On the consecutive trial (right), the features can be split between 1, 2, or 3 cards.

An ideal observer, as above, only takes two trials to reach the 100 percent criterion. A subject choosing randomly, would choose the rule feature at 25 percent. A subject that gradually attributes feedback would choose the rule feature with an increasing percentage as a function of the number of trials.

Monkeys and humans have drastically different behavior in choosing the single feature (Fig. 4.4(B-D)). Humans tend to choose the consistent feature nearly 100 percent of the time after $N = 2$ trials, consistent with the discrete memory model, hybrid memory model, and an ideal observers behavior (Fig. 4.4(C-D)). Monkeys instead increasingly choose the rule feature as a function of the number of correct trials (Fig. 4.4(B)), consistent with a reinforcement learning framework. In an RL framework, the belief that the common feature is the correct rule increases at a speed dependent on the learning rate. Further, behavior from the best fitting models match the subjects behavior in Fig. 4.4(B-D). With these statistics in mind, we believe that monkeys slowly build their belief towards the rule while humans are able to quickly perform correct inference.

Single Trial Inference

The ability to perform inference after a single trial is critical in learning this task. Differing computational frameworks will perform different inferences depending on the information received from feedback. We first analyze the probability of inference from correct feedback in the sequence of an initial incorrect trial followed by a correct trial. During the correct trial, the subject chooses three features, labeled as A, B, and C. On the trial following the correct trial, all three features could occur again on the same card. The three features could split, with 2 features on one card and 1 feature on another card. Lastly, the three features could split onto three separate cards. We label these cases ”1 card,” ”2 cards,” or ”3 cards” respectively, where the number refers to the number of cards features A, B, and C are split between. See Fig. 4.5 for a schematic detailing the feature splitting. We are interested in how often the subjects choose any card containing features A, B, and C depending on the three cases. An ideal observer would choose a card with features A, B, or C a hundred percent of the time. An observer choosing cards randomly, would choose a card containing one of the features at 25 percent chance for the one card case, 50 percent

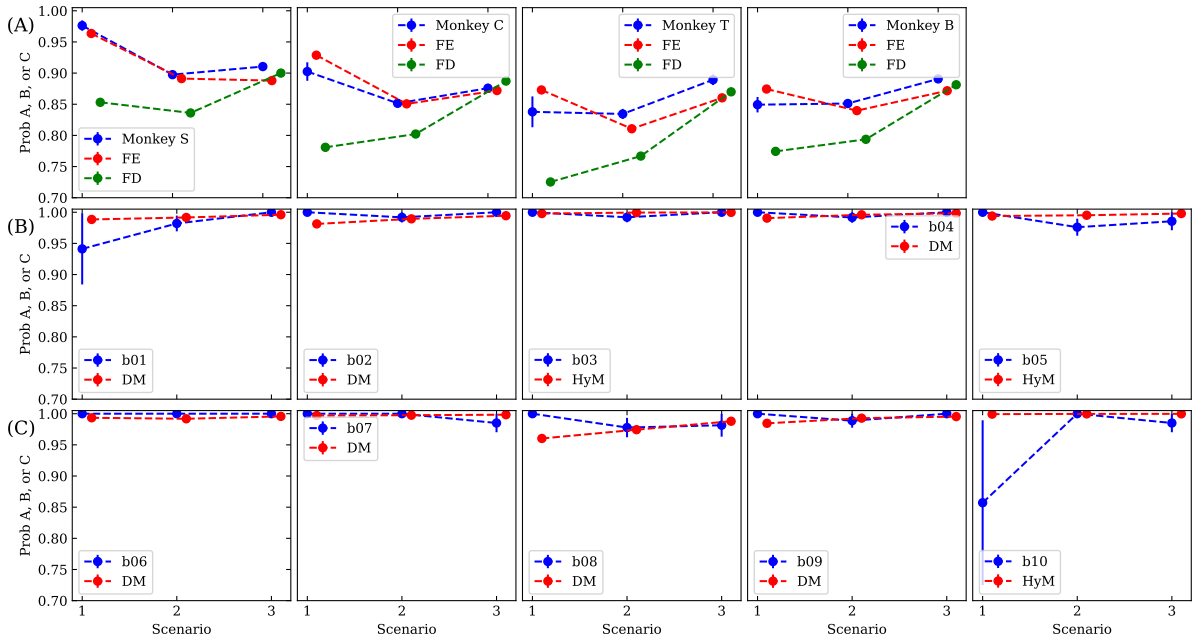

Figure 4.6: Plotting the likelihood of choosing any card with features A, B, and/or C on it when those features are all on one card, two cards, or three cards in monkeys (A), humans with instruction (B), and humans without instruction (C). After one incorrect trial, A, B, and C are the features from the correct trial, to be chosen on the next trial. For monkeys, we see a very high likelihood of choosing a single card with all three features. This occurs when the subject sees the same card after just choosing it. Human subjects tend to remain at 100 percent or close to it, showing an ideal agent like strategy. Fitting models and simulating best fit agents with the same parameters leads to similar behavior. We also simulate the Feature RL with Decay model (green), where it is unable to capture monkey behavior.

chance for the two card case, and 75 percent chance for the three card case.

A more subtle behavior is one with a linear addition of probabilities of choosing each feature. In this hypothesis, there is a length 12 hidden state vector $\mathbf{p}$ with probabilities of choosing each element: the probabilities of choosing a card k is the sum of probabilities $\mathbf{c}_k \cdot \mathbf{p}$. To simplify, the probability of choosing A, B, C, or a different feature is defined as p_A, p_B, p_C, and p_O respectively; the same conclusion is reached by allowing a different p_O per feature. If all three features are on one card, the chance of choosing that card is $p_A + p_B + p_C$. If the three features are split between two cards, the chance of choosing either card is $p_A + p_B + p_C + 3p_O$. If the three features are split between three cards, the chance of choosing any card is $p_A + p_B + p_C + 6p_O$. The probability of choosing a card with features A, B, or C grows linearly with the number of cards that the features are split between.

The probability of choosing a card containing the correct features depending on the number of cards is shown in Fig. 4.6. There are large differences between monkeys and humans. Humans typically follow ideal observer behavior, where they are very likely to always choose a card containing the previous correct features. The models, ran on simulated data and following ideal observer behavior, reasonably replicate human behavior. Monkeys, however, choose cards containing the correct features with a produced probability. Further, in monkeys S. and C., the probability is very high for the 1 card case. The probability is relatively equal between the 1 and 2 card case for monkeys T. and B.

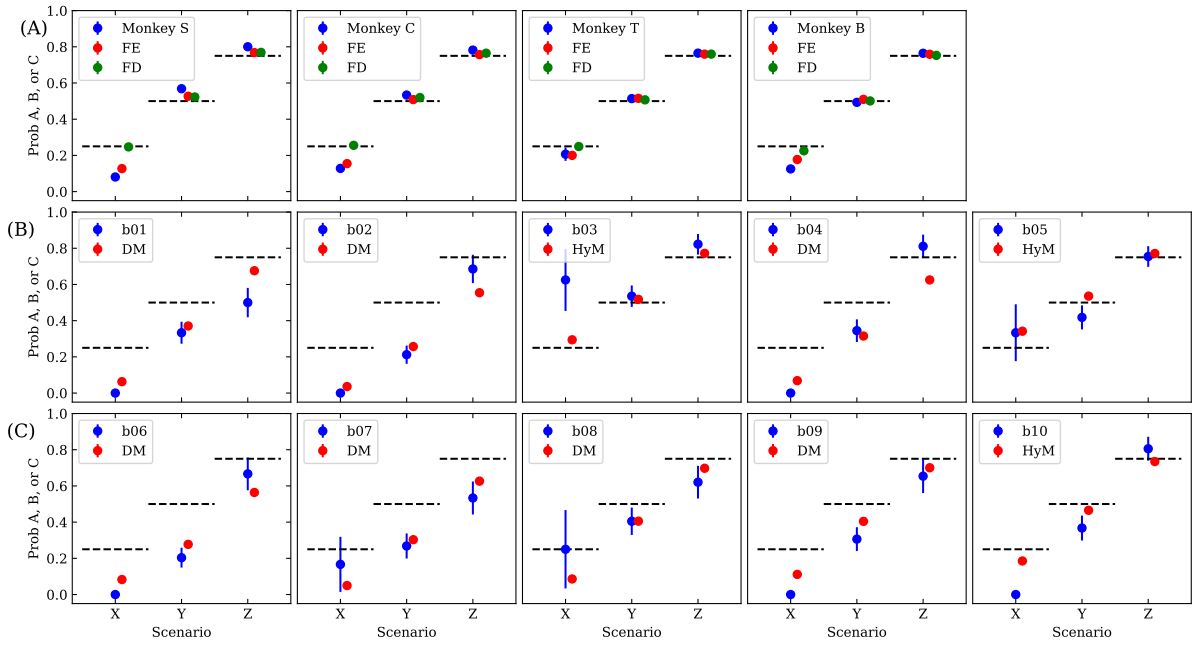

Figure 4.7: The likelihood of choosing any card with features A, B, and/or C when those features are on one card, two cards, or three cards in monkeys (A), humans with instruction (B), and humans with minimal instruction (C). After a correct trial and one incorrect trial, A, B, and C are features from the following incorrect trial; we are interested in the likelihood of choosing features A, B, or C after the second incorrect trial. Chance, given by a random agent's guess, is the horizontal dashed line. Monkeys avoid the three features if they appear on a single card, or when they see the repeated card. Humans avoid the three features at varying amounts. Some subjects avoid the three features at all times. Others avoid the three features only if they are split between one or two cards. Others do not avoid the three features. Simulations of best fit models have similar behavior to the majority of the subjects. We also simulate the Feature RL with Decay model (green), where it is unable to capture monkey behavior.

The monkey behavior is surprising and does not follow any of the standard hypothesized behaviors. They choose cards with the previously correct features less than an ideal observer but much higher than a randomly choosing observer. Further, the probabilities do not increase linearly with number of cards, which would be expected if the subjects chose cards based on summing the probabilities of choosing each feature. Instead, there must be a non-linear addition of probabilities of choosing each feature.

One possibility, motivated by the Feature RL with Decay model, is that the probabilities of choosing cards follows a softmax. Using $\mathbf{p}$ and defining $x_{\{1,2,3\}} = \exp(\beta(p_{\{A,B,C\}} - p_O))$, we can rewrite the probability of choosing a single card with all three features as $\frac{x_1x_2x_3}{x_1x_2x_3+3}$, the probability of choosing one of two cards with all three features as $\frac{x_1x_2+x_3}{x_1x_2+x_3+2}$, and the probability of choosing one of three cards with all three features as $\frac{x_1+x_2+x_3}{x_1+x_2+x_3+1}$. There are values for x_i that can lead to a larger probability of choosing the single card case than the other two. There are no possible real values of x_i to exactly match the probabilities of Monkey S. If p_O is allowed to be different for all of the other 9 features, it is possible to find a vector $\mathbf{p}$ that replicates the monkeys behavior. However, the necessary vector $\mathbf{p}$ would need to occur every time the sequence of incorrect to correct trials occur, which is unlikely. Further, we compare the behavior of the Feature RL with Decay model on simulated data in Fig. 4.6(A) (green), and it is unable to capture the high likelihood of choosing the repeated card.

The most likely hypothesis is that the Monkeys, mainly Monkeys S. and C., preferentially choose the card with the exact same card if it was correct. This is modeled by the extended feature RL model, where there is a boost in probability when seeing the exact same card that just received correct feedback. As seen in Fig. 4.6(A), this model is able to replicate the subjects average statistics. There are many individual sessions, in all NHP subjects, where the probability of choosing the same card when previously rewarded is 100 percent.

Inference from trials with negative feedback also is useful for identifying computational frameworks. Since negative trials are not technically needed to solve this task, it's possible to showcase widely different strategies when receiving negative feedback. We analyze the sequence of trials when the subject was correct followed by two incorrect trials; the first correct and incorrect trial are used to initialize the sequence since they most likely do not know the rule at this point. On the second incorrect trial, the subject chooses features A, B, and C. Like before, these features may appear on one card, two cards, or three cards. Ideal behavior is only fixed for the one card case, where the subject should avoid the card with all three negatively rewarded features. When the features are mixed, there may still be a benefit to choosing a card containing the previously incorrect feature due to the other features. However, on average, the likelihood of choosing any card with features A, B, or C should be reduced. As before, an observer choosing randomly would choose a single card containing the three features with 25 percent chance, two cards containing the features at 50 percent, or 75 percent chance if they lie on all different cards.

The probability of choosing features A, B or C on the following trial is shown in Fig. 4.7. There again is drastically different behavior for monkeys and humans. Monkeys choose these previous incorrect features like a randomly choosing observer for the 2 and 3 card case. They only avoid a card when all three previously incorrect features are on it. Further, we compare the Extended Feature RL model and Feature RL with Decay model on the monkeys, and only the Extended Feature RL model is able to capture the decrease in probability of choosing the repeated card. Humans instead are typically, with a few exceptions, able to avoid cards with the previously incorrect features on it. Again this shows the ability of humans to perform like an ideal observer.

From the behavioral statistics, we believe that the monkeys preferentially choose or avoid cards that contain all three of the previous correct or incorrect features. By simulating the best fit models on this task, the behavioral statistics of the models match the subjects (Figs. 4.6 and 4.7) for both correct and incorrect stimuli. In supplementary figures, we also analyze by session and by number of incorrect trials in the trial history. The amount of avoidance changes based on trial history. After only one incorrect trial, the subjects barely avoid the incorrect card.

4.4.3 Humans Typically Remember 2 to 3 Features

From the modeling fits in Fig. 4.2, the best fit models have a substantial memory limitation, implying that humans only remember a few features at a time. Focusing on subjects b01, b02, b04, b06, b07, b08, and b09, the best fit models are when $N_C = 1$ (23% of the time), $N_C = 2$ (63% of the time), $N_C = 3$ (14% of the time), $N_I = 0$ (9% of the time), $N_I = 1$ (23% of the time), $N_I = 2$ (14% of the time), $N_I = 3$ (23% of the time), $N_I = 4$ (17% of the time), $N_I = 5$ (9% of the time), $N_I = 6$ (3% of the time), and $N_I = 10$ (3% of the time). When applying this strategy, humans tend to only remember 2 out of the 3 features from a correct trial. This is quite surprising given the humans have routinely shown the ability to remember more than 2 things [84, 85]. However, it's plausible since there is a lot of stimuli shown to them that they could easily become distracted.

We further analyze the memory limitation by examining the ability for human subjects to remember all three features after a correct trial. Through behavior, we check how often the human subjects remember all three features when they are gradually ruled out through trial and error. We initialize the trials with an incorrect trial followed by a correct trial with chosen features A, B and C. On the following trial, we ask how likely the subjects are to choose any of the three features (top branch of Fig. 4.8(A)). In Fig. 4.8(B), the left column shows that all 10 subjects are extremely proficient at selecting a card containing at least one of these features. Next, we check if one of the features is ruled out. After the correct trial, and a following incorrect trial that ruled out feature A, we check how often the subject chooses either feature B or feature C (middle branch of Fig. 4.8(A)). In Fig. 4.8(B), the middle column shows that about half of the subjects perfectly remember at least two features. A few of the subjects

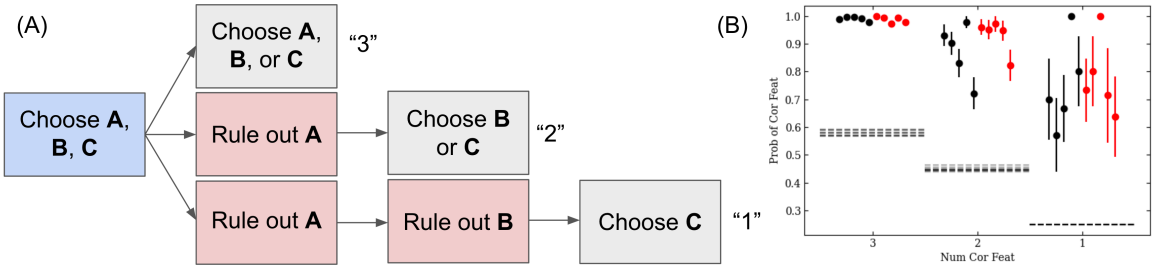

Figure 4.8: The probability of choosing remaining features after ruling out features with intermediate incorrect trials. (A) shows a schematic of the three possibilities. After an incorrect trial (not shown), there is a correct trial (in blue), where the subject chooses features A, B, and C. The different branches represent the three cases of interest, where single features are ruled out with incorrect trials (red). In the final trial (grey) of each branch, we ask how often the subjects choose a card with the remaining number of features. The likelihood of choosing one of the remaining features is plotted in (B), where each branch directly corresponds to the same labeled column. Dots in black represent humans with instruction and dots in red represent humans without instruction. Horizontal dashed lines indicate, for each subject, the average probability of choosing one of the remaining features given a random choice. The dashed lines occur in their respective locations due to the number of times each of the three features end up on one card, two cards, or three cards, as in Fig. 4.5. In many subjects, there is a distinct drop off in performance after two incorrect trials, which indicates they don't always remember all three features. Other subjects have high performance, such as b04 or b08, and remember all three features.

choose features B or C with less than ideal behavior. Lastly, we check if two features are ruled out, how often the subjects choose the third feature (bottom branch of Fig. 4.8(A)). In the right column of Fig. 4.8(B), we see that 8 out of 10 subjects do not choose the third feature with ideal accuracy. However, all 8 subjects have accuracy above chance.

The main result from Fig. 4.8 is that the memory of all three features is typically much less than in the ideal strategy but not quite as low as a random strategy. This implies that humans can remember all three correct features, however they do not always do so. This may explain why the best fitting discrete memory model does not typically have $N_C = 3$.

4.5 Discussion

Our results suggest vastly different strategies for solving high dimensional credit assignment problems between monkeys and humans. By comparing the one step prediction accuracy and behavioral statistics of a variety of models, we find that monkeys tend to follow a reinforcement learning strategy while preferentially selecting or avoiding previously correct or incorrect cards. Monkeys tend to gradually learn the correct rule while gradually forgetting non-chosen features. They are slow to ignore incorrect features as they tend to incorrectly choose previous rule features for much longer than expected. Humans on the other hand tend to showcase limited memory ideal-observer behavior, where they select cards that maximize reward based on what is held in memory. While they have the ability to remember three correct features, they sometimes forget 1–2 features. They are quick to learn the rule and quick to forget the

previous rule. Thus, the strategies and computational frameworks are vastly different per species.

For monkeys, the best fitting model is one with reinforcement learning and an added naive component that biases the choice towards/away from previously correct/incorrect cards. This further provides evidence for the behavioral result of monkeys remembering the previous choice. The question remains as to why the bias from the previous choice occurs in monkeys but not humans. It's possible that the humans do not need to bias their decisions due to high task performance. It would be interesting to observe the bias in tasks with a spectrum of difficulties. When there is non-perfect task performance, it may be most efficient to remember the previous trial as opposed to relying on evidence from incremental learning. Further, the effect could depend on fraction of occurrence. Given four cards in this task, each with 64 possible combinations of features, the previously chosen card repeats an average of 1 out of 16 times. If the fraction of occurrence increases/decreases, it may be more/less beneficial to remember the previous trial.

Another remaining question is why monkeys, unlike humans, are unable to perform like an ideal observer. One may assume that monkeys would Bayesian optimally solve the task if there were less objects and features. For example, monkeys Bayesian optimally solve two arm bandit reversal tasks [53, 158]. However, even in a simpler but similar task, monkeys have shown hybrid Bayesian and incremental behavior [83]. It would be interesting to test monkeys with varying number of objects and features to see when monkeys switch behaviors from ideal to incremental learning. The same switch between ideal and incremental may be present in humans as well. While memory-limited ideal in this task, when adjusting the feedback to be non-deterministic [94], humans switch to a reinforcement learning strategy. It's possible that a spectrum of difficulties would reveal a change point in strategies from optimal to sub-optimal performance.

One major difference between humans and monkeys is the instruction. There is no verbal instruction for monkeys, and monkeys learn by trial and error. There are two sets of humans with differing instruction. In one group, the verbal instructions explain the task. In the other group, verbal instruction is minimal to mimic the monkeys. This latter group is told that they will sometimes be correct or incorrect and that the goal is to maximize the number of correct responses. While not exactly the same, it's similar in idea to how the monkeys learn. Despite the human instruction difference, the inter-subject difference is much larger than the inter-instruction difference. We believe that the difference in strategies is more likely to come from internal computational frameworks instead of purely instruction.

Lastly, this works allows for future work in comparing neural data with hidden values. It is exceptionally difficult, without behavioral modeling, to understand what subjects are thinking when they choose a card on any trial. E.g., if a subject chooses a red striped triangle, did they choose this card because it was red, striped, triangle, or some combination of features? The models here provide a framework to understand this question. One can look at the hidden values and choices for each trial to gain insight

into the most likely feature of interest. Then, when comparing to neural data, one may be able to find representations of hidden values or even beliefs of particular rules. One can also calculate quantities such as reward prediction error [64, 159, 160] from the probabilities of choices. Thus, this work provides the foundation for future work on neural representations during this task.

4.6 Appendices

4.6.1 Non-Human Primate Experimental Details

Four adult female rhesus monkeys (Macaca Mulatta) were used in this study, labeled as Monkeys S, C, B, and T. Their age at the start of the recordings was 9.1, 10.8, 19.9, and 10.3 years respectively. Their weights at the start of the recordings was 7.9, 7.6, 8.5, 5.9 kg respectively. was 12.5 $\pm$ 2.5 years at the time of the recordings, and they have a weight of 7.5 $\pm$ 0.6 kg. The subjects completed 42, 29, 89, and 23 sessions respectively, with 959 $\pm$ 269 trials per session. Evaluated by the experimenter, every 31 $\pm$ 5 minutes, the subjects rested for 12 $\pm$ 3 minutes while seeing a blank screen. Programming occurred with NIMH Cortext or NIMH MonkeyLogic software. All subjects were equipped with eye tracking (EyeLink 1000 Plus, SR Research), and all subjects were head-fixed with a titanium rod. Monkeys S and C had microdrives implanted for corresponding neural analysis. All procedures were carried out in accordance with the National Institutes of Health guidelines and were approved by the University of Washington Institutional Animal Care and Use Committee.

Each NHP was head-fixed in a dimly illuminated room, positioned 60 cm away from a 19-inch CRT monitor with a resolution of 800 $\times$ 600 pixels, corresponding to 33 degrees by 25 degrees of subtended visual angle.

Several bugs occurred throughout the sessions. Some of Monkey S's sessions had single-trial rule blocks. Some of Monkey C's sessions had rule switches for any number of correct trials between 7 and 16.

4.6.2 Human Experimental Details

There were a total of 10 subjects, referred to as b01 to b10, recorded in the Knight lab. Each subject performed 5 sessions in total, with 300 trials per session. Subjects b01 through b05 were given instruction on the task, while the others were given minimal instruction to mimic the instruction given to the monkeys. For subjects b01-b05 there were 4 adult males and 1 adult female with a mean age of 26.4 $\pm$ 4.1 yrs. For subjects b06-b10 there were 2 adult males and 3 adult females with a mean age of 20.2 $\pm$ 1.6 yrs.

The instructions given to subjects b01-b05 is: "in this experiment, you will see 4 cards on each trial. Each card has 3 unique features (color, shape and texture). The card containing the correct feature will be the correct choice. The correct feature might change during the task. Press any key to continue. The answer is given by pressing one of the four arrow keys that corresponds with the selected card position on the screen (up, down, left or right). You have 4 sec to provide the answer, or the trial times out. Good luck and thank you!" Note that "texture" is the same as what we refer to as "pattern" in the manuscript. The instructions given to subjects b06-b10 is: "In this experiment, you will use one of the four arrow keys on each trial as a response. The 'correct' or 'incorrect' feedback will be provided following each choice. Your task is to maximize the number of correct responses."

The studies involving human participants were reviewed and approved by the Institutional Review Boards of University of California, Berkeley. All participants provided their written informed consent to participate in this study and received a small compensation.

4.7 Supplementary Material

4.7.1 Evaluating Models Based on Numbers of Parameters

In the main text, we found the models that best predicted the subjects choices via a negative log-likelihood calculation. One may complain that models with extra parameters will naturally fit better. To compensate, we turn to the Akaike Information Criterion (AIC) and Bayesian Information Criterion (BIC), which penalize fits based on number of parameters.

Using the likelihood L from Eq. 4.34, the AIC is defined as:

$$\mathrm{AIC} = 2k - 2\log(L), \tag{4.37}$$

where k is the number of parameters. However, given that the L is quite large due to the high number of samples, the AIC is barely affected by the number of parameters. This is shown in Fig. 4.9, where the same trends are seen as the main text.

The BIC is defined as:

$$\mathrm{BIC} = k\log(n) - 2\log(L), \tag{4.38}$$

where k is the number of parameters, and n is the number of samples. In this case, the BIC can be affected highly from the number of parameters, since both terms are scaled with the number of samples. In Fig. 4.10, we see the monkey fits minorly affected and no difference in the human fits. In two of the monkeys, the Extended Feature RL model performs equivalently with the Feature RL model with Decay. As shown in the main text, behavioral metrics lead to a better performance by the Extended Feature

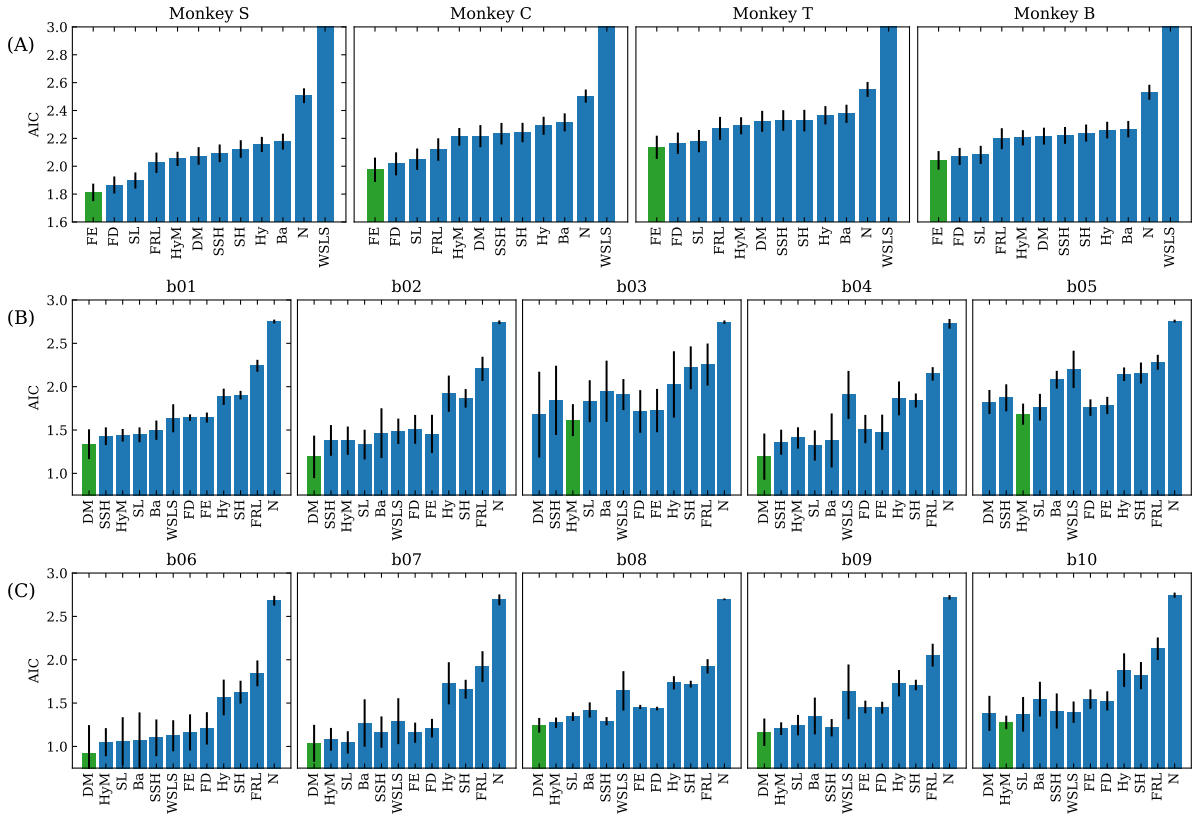

Figure 4.9: Average AIC performance (AIC/N) for models fit to non-human primates (row A), humans with instruction (row B), and humans with minimal instruction (row C). Models are sorted by Sam in (A), subject b01 in (B), and subject b06 in (C). The best fit models are in green. The best performing models are the same as that in Fig. 4.2.

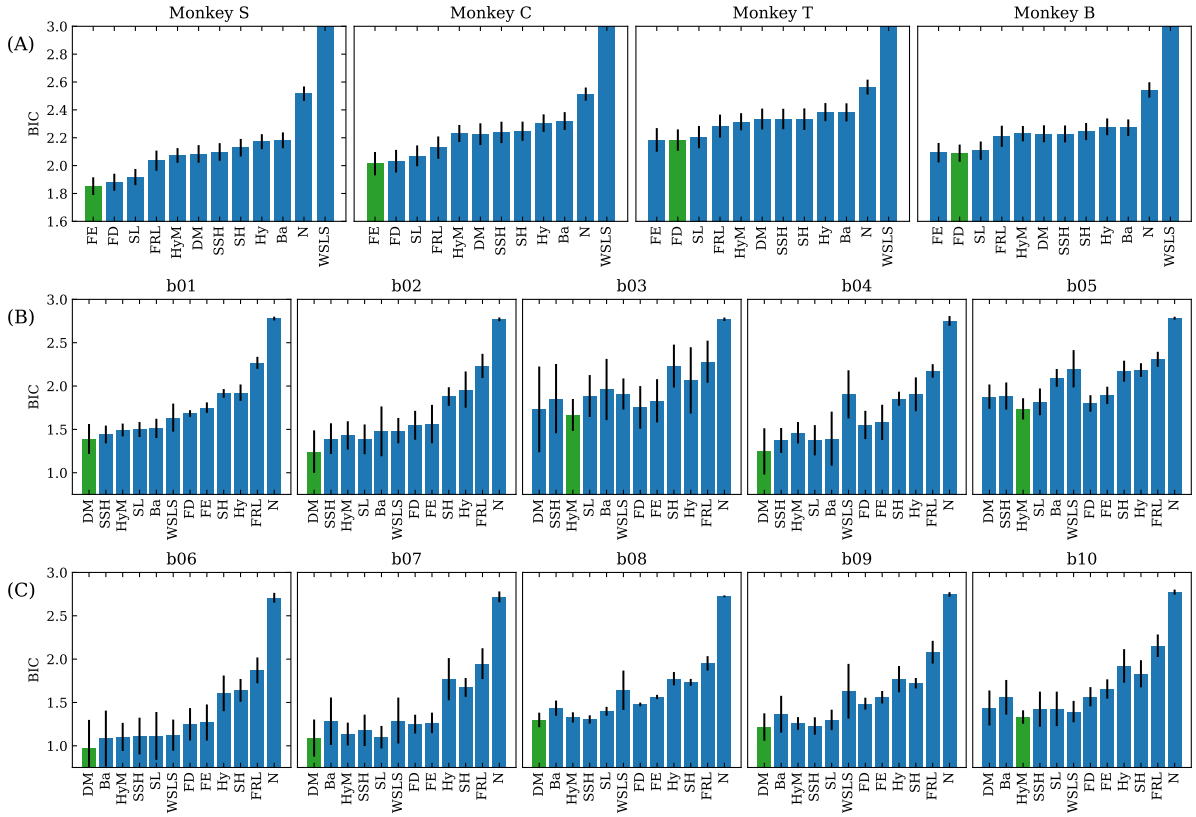

Figure 4.10: Average BIC performance (BIC/N) for models fit to non-human primates (row A), humans with instruction (row B), and humans with minimal instruction (row C). Models are sorted by Sam in (A), subject b01 in (B), and subject b06 in (C). The best fit models are in green. The best performing models are nearly the same as that in Fig. 4.2. However, the BIC performance is very similar between the extended feature RL and feature RL with decay models.

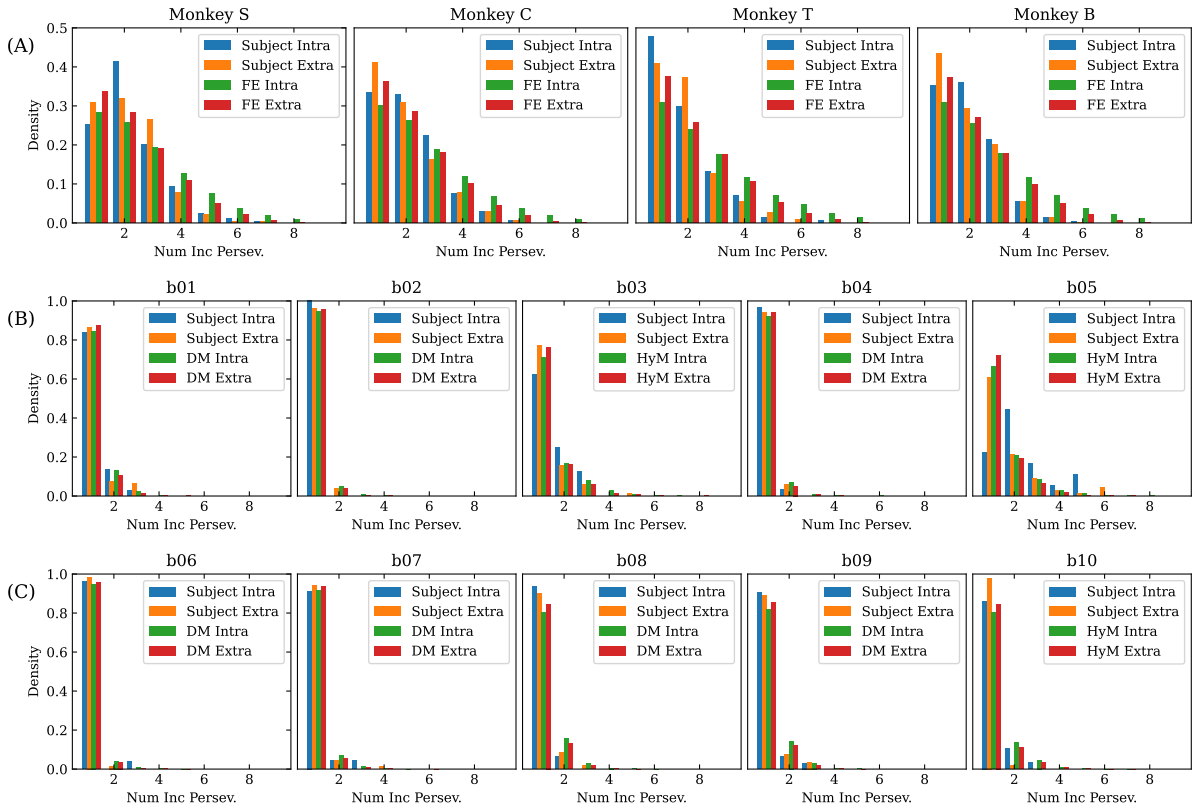

Figure 4.11: The perseveration, or choice of the previous rule feature, of non-human primates (row A), humans with instruction (row B), and humans with minimal instruction (row C). After a rule switch, given a minimum of one incorrect perseveration, the x-axis is the number of incorrect perseverations that occurred before choosing a card without the previous rule feature on it. Monkeys perseverate for a few trials. Humans typically either avoid the previous rule feature or switch to a random choice after receiving incorrect feedback. Each best fit model is simulated on the task, and the statistics match the subjects behavior.

RL model.

4.7.2 Intra vs Extra Dimensional Shifts

Depending on if the rule switched feature dimensions (extra dimensional shift) or maintained the same feature dimension (intra dimensional shift), the post rule-switch statistics are distinct. During an extra dimensional shift, it's possible to choose both the new and old rule feature during extra dimensional shifts but not intra dimensional shifts. For this reason, we treat both shift types independently.

With the exception of subject b05, there is no major difference between intra and extra dimensional shifts (Fig. 4.11). Subject b05 seems to learn faster from extra dimensional shifts. To analyze further, we check the speed that each rule is learned depending on the shift type. We compute a two-sided independent t-test with an alpha value of 0.05 for the number of trials to criterion across all shift blocks. The p-values for each subject are 0.138 (Monkey S), 0.373 (Monkey T), 0.003 (Monkey C), 0.048 (Monkey B), and for subjects b01 through b10: 0.27, 0.464, 0.299, 0.361, 0.591, 0.47, 0.854, 0.006, 0.078, 0.08. The null hypothesis is rejected for human subject b08, where they solved intra shift trials in 19.4 trials

and extra shift trials in 14.8. The null hypothesis is rejected for Monkeys C. and B. The difference (intra minus extra) in mean trials to criterion is 1.8 (Monkey S), 2.7 (Monkey T), 3.4 (Monkey C), and 2.6 (Monkey B). The difference in trials to criterion most likely occurs due to accidental positive rewards while perseverating during extra dimensional shifts, which cannot occur in intra dimensional shifts. Overall, there is no drastic change in performance across 9 out of 10 human subjects, and the differences in all monkeys and the 10th human most likely occurs due to chance during extra dimensional perseverative effects. We don't believe that humans or monkeys prefer focusing on dimensions as opposed to individual features.

Chapter 5

Neural Correlates of Behavioral Models

The behavioral models in Chapter 4 help explain the subjects choices on every trial. These models have a hidden value associated with every feature, which are used to calculate the probability of choosing any particular card. The hidden values lead to an understanding of the subjects internal state, corresponding to the subjects belief of the world.

For monkeys, the best fit models are reinforcement learning models with an additional memory of the previous correct or incorrect card. They gradually build up evidence towards individual features as they receive correct feedback. As an example, the hidden values increase for one particular feature throughout the rule block (Fig. 5.1). The hidden value remains large for a few trials after the rule changes, correlating to perseveration. During the middle of each block, multiple features can simultaneously have large associated hidden values, allowing the subject to focus on multiple features simultaneously.

While there is behavioral evidence for the reinforcement learning model, neural evidence is needed to validation. Are there representations of one or multiple features? Are there correlations to hidden values? The hidden values are continuous, are the neural representations continuous? Is there a memory of the previous stimulus, as implied by the best fit behavioral models?

There are multiple ways to tackle the above questions. While one can directly regress neural firing rates to the hidden values, one can be more clever with the interpretation of the hidden values. During every trial, the subject chooses three features. Assuming a non-random choice, the largest hidden value of the three features is the most likely feature that the subject is ”focusing” on, which can be thought of as the subject ”belief.” Towards the end of the rule block, typically only one feature has a large hidden value. Towards the beginning, all three features typically have small hidden values. This approach allows the evaluation of belief even at the beginning of the block. However, this approach neglects the possibility of multiple features simultaneously influencing the choice.

A more general approach is to aggregate the hidden values across features. Towards the beginning of each block, the hidden values tend to be small. Towards the end of each block, there tends to be

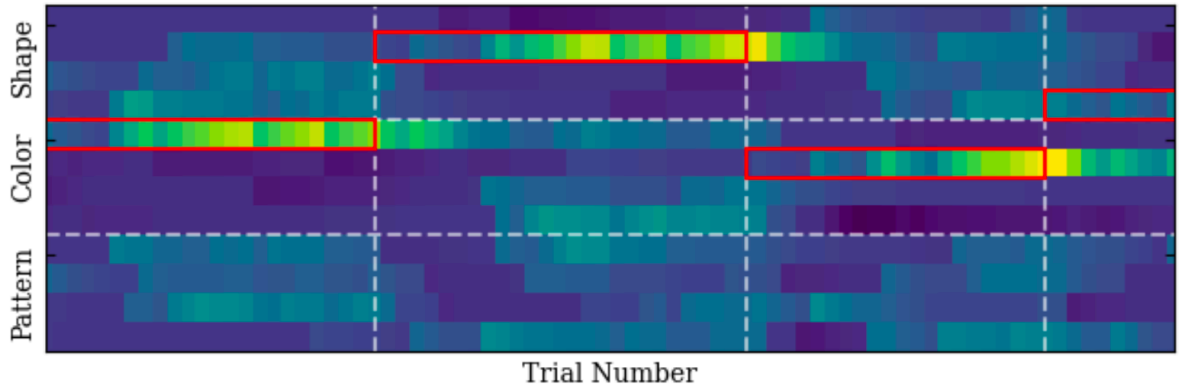

Figure 5.1: Example hidden values across trials for Monkey S. Yellow indicates large hidden values. Each row is a different feature. Horizontal dotted lines separate feature dimensions. Vertical dotted lines separate blocks. Red boxes indicate the true rule. The hidden values increases when the subject receives correct feedback for a feature. The rule features associated hidden values increase and remain after the rule switches.

one feature with a large hidden value. These two extremes allow for a measure of "confidence," where many small hidden values indicate low confidence and several large hidden values indicate high confident. Aggregating across hidden values also allows for the use of reward prediction error (RPE), which has been shown to correlate to activity from dopamine neurons [64, 159, 160]. RPE is calculated as the probability of correct choice compared to the true reward value of 1 (correct) or 0 (incorrect), where the probability is calculated from all cards [1].

To find neural evidence for the best fit behavioral model for monkeys, the extended feature RL model (Sec. 4.3.2), I perform several key analyses. Direct regressions to the hidden values shows that some neurons have a small correlation to the hidden values. From direct observation of single features, the neural firing rates show differences based on the amplitude of the associated hidden values. This result is confounded by the observed card, which is then supplemented with a generalized linear model (GLM) approach. GLMs improve the fitting of working memory when model hidden values are included. To explore further, decoding approaches can directly find neural populations responsible for encoding the "belief" of the monkey after receiving feedback, a necessary representation for updating the internal state. Lastly, I focus on representations of previous correct cards, where there are differences between choices of repeated and non-repeated cards.

This chapter focuses on neural spiking data, binned in 100 ms bins with a 50 ms stride around the object fixation and feedback period.

5.1 Regressing to Behavioral Model

Is there a linear relationship between neural firing rates and the models hidden values? If so, this provides the most obvious interpretation of a particular neurons role: there would be particular feature neurons

[1] Patrick Zhang has begun exploring the use of RPE and has found neural differences based on RPE

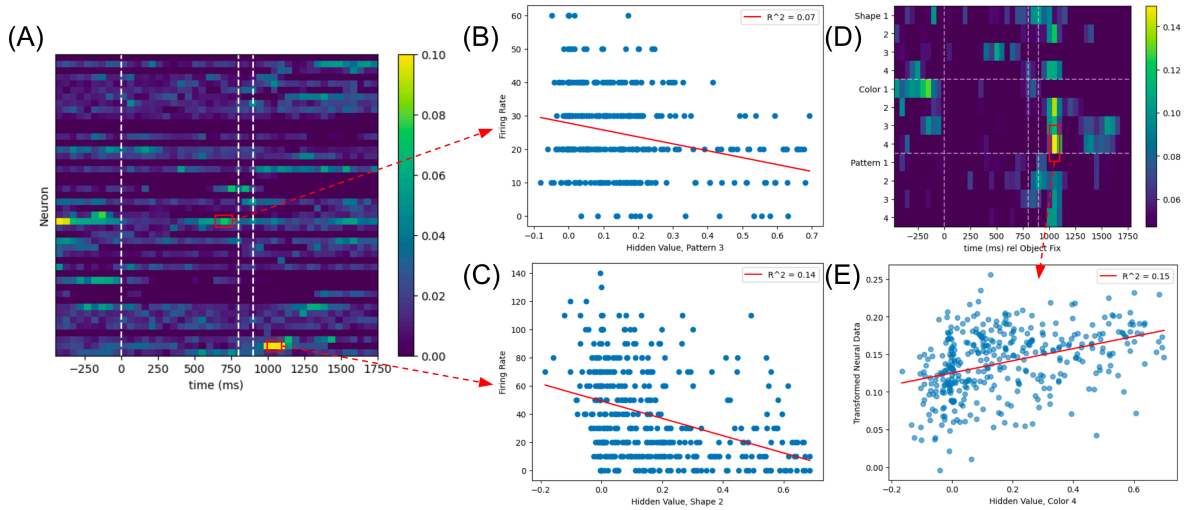

Figure 5.2: (A) Maximum coefficient of determination (R^2) across features, calculated independently for each timepoint and neuron in session 2018-08-02. Trials must include the feature of interest, and at least 10 percent of trials need to have a non-zero firing rate. Vertical dotted lines in all plots refer to the beginning/end of object fixation and feedback onset (B/C) Example linear fits for particular neurons and timepoints (arrows from (A)), showing the firing rates and hidden values for an example feature. (D) For the same session, R^2 between hidden values and transformed neural data for particular timepoints when transforming the firing rates based on best fit regressions from all timepoints. Color scale begins at five standard deviations above the control mean, which is calculated for each timepoint with shuffled hidden value labels. (E) Example scatter-plot of hidden values for the transformed data at a particular time point.

that focus on particular features. However, it's possible that more complex relationships exist, such as non-linearities or neural sub-spaces that encode hidden values. This section focuses on two analyses: linear decoding of firing rate from hidden values and population encoding of hidden values.

To examine any possible linear relationships, a linear model is calculated to predict the firing rate from the hidden values for each feature, timepoint, and neuron. Trials are included if the subject chose the feature of interest, and at least 10 percent of trials must have a non-zero firing rate. Linear models are fit and evaluated with the coefficient of determination (square of the Pearson coefficient R^2). While possible for neurons to be tuned to specific features, information is down-sampled by taking the maximum R^2 across all 12 features. The coefficient of determination (Fig. 5.2(A)) shows that the best fit neurons have a weak relationship. Across all sessions and times, including the cross fixation and object fixation, best fit relationships have a value of $R^2 \approx 0.2$. That being said, the weak fits show some trends, where the linear models sometimes fit better before object fixation, during object fixation, or during feedback.

Two neurons and timepoints are selected for further analysis, one during object fixation and another immediately after feedback. The features with maximum R^2 are selected for the respective neurons. The firing rates and hidden values for each trial (Figs. 5.2(B,C)) show the weak linear relationship. While these neurons were negatively correlated, other neurons show positive correlations.

Does a population analysis lead to better correlations?Specifically, is there a population subspace that linearly encodes hidden values? To examine this, linear models are found to predict the hidden values from the population of neurons. Since there are many neurons and relatively few trials per feature,

timepoints are concatenated. That is, a linear model for a particular feature is fit based on the firing rates for all timepoints within all trials where the subject chose that feature. This linear model is used to transform the neural data at every timepoint, and the transformed data is linearly regressed to the hidden values.

The coefficient of determination is calculated for each feature and timepoint of the transformed data in Fig. 5.2(D). As a control, the same analysis is done 10 times with shuffled hidden values. The color scale in Fig. 5.2(D) begins at the five standard deviations above the controls mean R^2. Many features show an above chance linear relationship immediately after feedback. There are also features with an above chance linear relationship before object fixation, during object fixation, and well after feedback. One feature, immediately after feedback, is examined further in Fig. 5.2(E). The transformed data is regressed against the hidden values. Compared to Figs. 5.2(B,C), the linear relationship is much more clear, which is the case in all other fits. Despite relatively small correlations, peaking at $R^2 \approx 0.15$, these plots are more convincing of a linear relationship.

From this section, there is a weak correlation between the hidden values and neural firing rates. Using population methods, the neural firing rates can be transformed into a representation with a more convincing linear relationship. While the weak correlation could be due to neural variance being explained by other behavior, it's possible that the exact neural representation of the behavioral model may be non-linear. At present, it is unclear what the non-linear relationship may be.

5.2 Firing Rate Differences

The simplest question one can ask is whether units respond differently depending on the belief of the subject. On every trial, three features are simultaneously chosen, and the feature with the largest associated hidden value is the most likely to be the belief of the subject on that trial. Is there a difference in firing rate when the hidden value is large or small? Trials are selected if feature A was chosen, and the trials are divided into two categories: trials where feature A has a large associated hidden value and the other two features have small associated hidden values, and trials where A has a small associated hidden value and at least one of the other two features has a large associated hidden value. Large and small are defined as either above or below the mean hidden value. The mean differenced firing rate is found by subtracting the mean firing rate for both conditions (large minus small). The standard error of the mean is calculated as $(\sigma_1^2/N_1 + \sigma_2^2/N_2)^{1/2}$, where N is the number of samples. The process is repeated across neurons for every feature. Differences for an example neuron are shown in Fig. 5.3.

To calculate significance, two controls are calculated independently for every timestep. Poisson distributions are defined based on the mean firing rate. The Poisson distribution is sampled 50 times. The second control is based on the problem of a slowly drifting firing rate: the same process is followed,

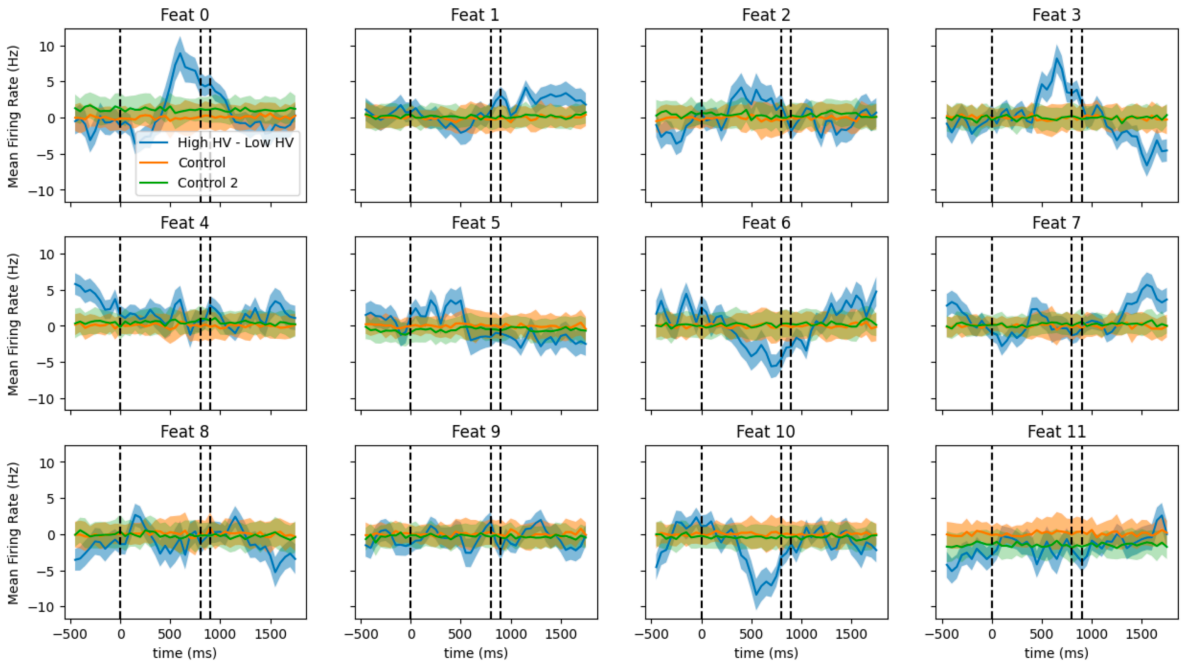

Figure 5.3: A random units firing rate differences for each feature when that feature has the largest hidden value compared to the smallest hidden value. Each row is a different dimension. The first dotted line indicates fixation on the choice, and the last dotted line indicates when feedback begins. The orange control is based on random firing rates from a Poisson distribution defined by the mean firing rate. The green control is based on random firing rates calculated from a Poisson distribution defined by the running average firing rate (10 trials).

but the mean of the Poison distribution is calculated based on the average firing rate of 10 consecutive timesteps. The mean for the first and last 5 trials are calculated by repeating the firing rate for the first and last trial.

There are many example features that show significant differences during the object selection period (Fig. 5.3), and many units show differences during all parts of the task (not shown). This indicates that some neurons represent belief towards particular features. Specifically, there are differences in firing rates when particular features have the highest chance of being selected.

With Fig. 5.3, features across dimensions are linked together. E.g. on some trial feature 0 may have the largest associated hidden value but feature 10 has the lowest associated hidden value. This trial will be included in the averages calculated for both features 0 and 10 in Fig. 5.3. It's plausible that as one features mean differenced firing rate increases, another features mean differenced firing rate may decrease. While this is divided across multiple features, some information is over-represented. That being said, the results here are indicative of feature information being present, regardless of which features are represented by a neuron.

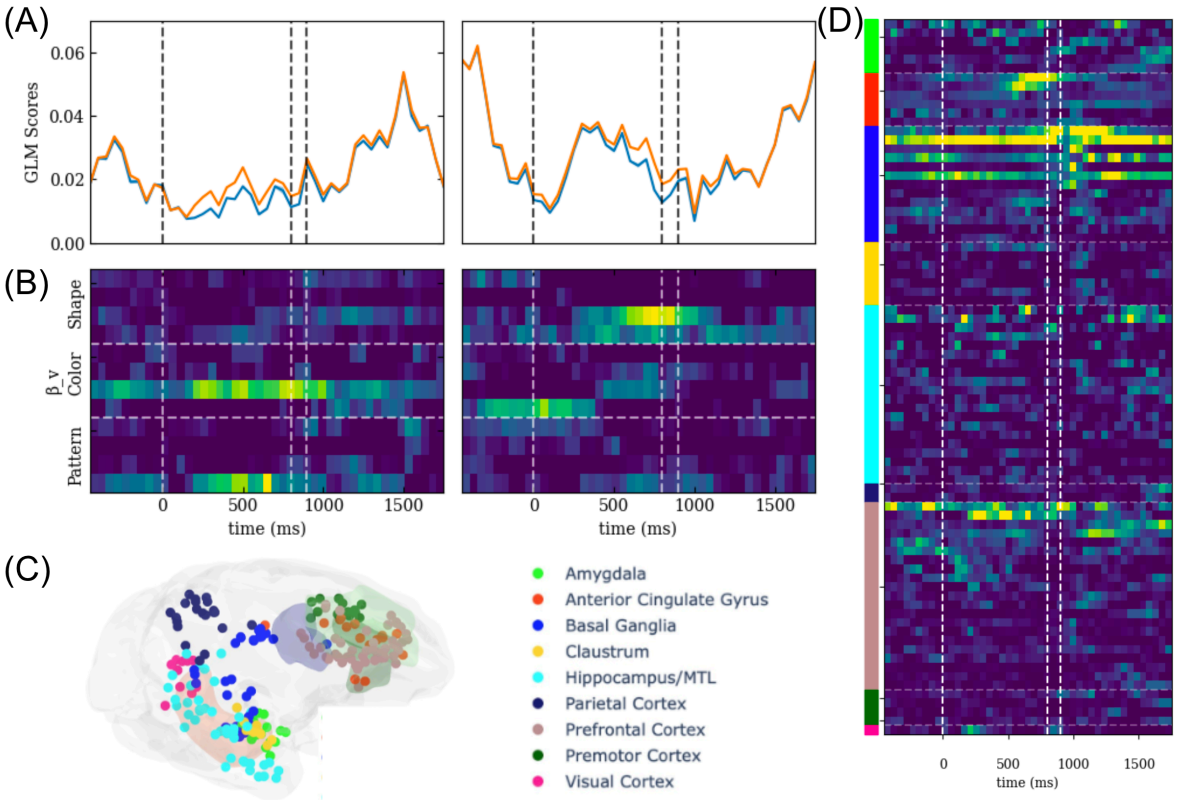

Figure 5.4: (A) Example units GLM scores when including model beliefs (orange) compared to random model beliefs (blue). GLM scores are calculated as the percentage of deviance explained. (B) Corresponding values for β for each dimension in the GLM fits (blue in (A)). (C) Approximate locations of all neurons. (D) GLM Score improvements (model beliefs - control) for threshold neurons (> 0.0015), sorted by maximum improvement per brain area. Vertical dotted lines refer to the beginning/end of object fixation and feedback onset.

5.3 Generalized Linear Model Representation

In the previous section, trials are used across features, making it difficult to understand how the firing rate correlates with the model. In an approach similar to that used by Pillow et al. [161], generalized linear models (GLMs) lead to feature information. In this approach for every trial, firing rates are modeled by a combination of all chosen features, feedback, previous feedback, and belief of the rule. The additional rule belief leads to better representations, as shown in the GLM fits, of the firing rates. GLM model scores are calculated with the percentage of deviance explained, which compares the deviance [7] of the calculated firing rates with the deviance from the average firing rate.

The firing rate r for a particular neuron at a particular time is modeled as

$$\log(r) = \beta_{\text{stim}} \cdot \mathbf{d} + \beta_{\text{FB}} \cdot \mathbf{FB} + \beta_{\text{v}} \cdot \mathbf{v}, \tag{5.1}$$

where β refers to the parameters of the model, $\mathbf{d}$ refers to the decision of the subject, $\mathbf{v}$ refers to the hidden values, $\mathbf{FB}$ refers to the current and previous feedback, and $\odot$ refers to the Hadamard or element-wise product. Using a log function to convert the firing rate is standard practice [161] due to the Poisson like nature of the firing rate. This also allows for the positive firing rate to match any values on the right-hand side. If a linear model works better, the parameters β will implicitly scale to a linear regime. A bias term is also included in the GLM. One can imagine more terms in the GLM, such as nonlinear mixed terms [2]. However, these three terms provides a decent beginning to the GLM analysis.

The first term in Eq. 5.1, $\beta_{\text{stim}} \cdot \mathbf{d}$, refers to the three features that are chosen on any given trial. Since $\mathbf{d}$ is zero when a feature is not chosen and one when chosen, β_{stim} models stimulus representations. During the object fixation period, β_{stim} primarily encodes neural firing rates that represent the visual stimuli. During feedback, no visual stimuli exist, so non-zero β_{stim} would be related to some memory of the seen features.

The second term in Eq. 5.1, $\beta_{\text{FB}} \cdot \mathbf{FB}$, refers to the combination of current and previous feedback. There are 2×2 cases that can occur: when the previous feedback is correct/incorrect and when the current feedback is correct/incorrect. One may expect that previous feedback is prominent early and current feedback is prominent later in the trial, but other effects may occur. Interaction terms can be represented if both previous and current feedback occur simultaneously. It's also possible for a neural response to current feedback the subject receives feedback, which may indicate the confidence of their choice.

The last term in Eq. 5.1, $\beta_{\text{v}} \cdot \mathbf{v}$, refers to the belief of a particular rule. The hidden values $\mathbf{v}$ relate to the probability of choosing any feature, so a non-zero β relates to the neural firings responsible for

[2] proposed by Patrick Zhang

encoding the belief. There are other methods that one can use. In particular, using one-hot encodings of belief, where $\mathbf{v}$ is set to one for the feature with the largest associated hidden value, works well in explaining the firing rate (not shown). The approach taken here has the benefit that all information is included from the behavioral model.

To evaluate whether the behavioral model helps model the firing rates, a control is needed with no information from the behavioral model. To avoid overfitting, the control uses the same number of parameters as the aforementioned GLM model. With this in mind, the control shuffles the hidden values $\mathbf{v}$ across trials. The relative hidden values across features remain the same. This way, any bias towards features will remain.

The GLM scores for two example neurons are shown in Fig. 5.4(A), where performance increases during the object search period. There are neurons with an increase in performance during all regions of the task. The small GLM score, compared to a best fit of 1, indicates that the overall ability to model firing rates with GLMs is fairly poor. That being said, the GLM fits do improve when the belief of the rule is included.

The corresponding GLM parameters β are shown for the hidden values in Fig. 5.4(B). The largest weights correspond to one color and one pattern for the first neuron and two shapes and one color for the second neuron, indicating that neurons may represent multiple features simultaneously. Many neurons also have large parameters for the stimulus features and the feedback (not shown). For example, the neurons (Fig. 5.4(B)) both have a large weighting when the previous trial was correct up until the current feedback and large weighting for the correct trial after feedback occurs. The right neuron also shows selectivity for one of the shape features at the same time as the belief of the shape feature. One neuron (not shown) shows large feedback weighting for the correct trial and previous incorrect trial simultaneously during the feedback period. There is a rich amount of selectivity for each feature, feedback, and belief in most neurons.

To evaluate further, the GLM score differences (model minus control) are found for all neurons, showing evidence for better predicted firing rates with the inclusion of the behavioral model. Locations of the neurons are shown in Fig. 5.4(C). The score differences (Fig. 5.4(D)) are sorted and thresholded by the maximum score difference > 0.0015. The most prominent area is the basal ganglia (blue), where there are neurons that respond very clearly immediately after feedback and some neurons respond consistently for the duration of the trial. There are also increases in GLM scores in the PFC and anterior cingulate during the object fixation and feedback period. The Claustrum shows some general trends of GLM score increases during the feedback period. Many areas show some sporadic score increases, such as the medial temporal lobe and hippocampus. From this, the hidden values of the behavioral model play a key role in predicting the firing rates of neurons.

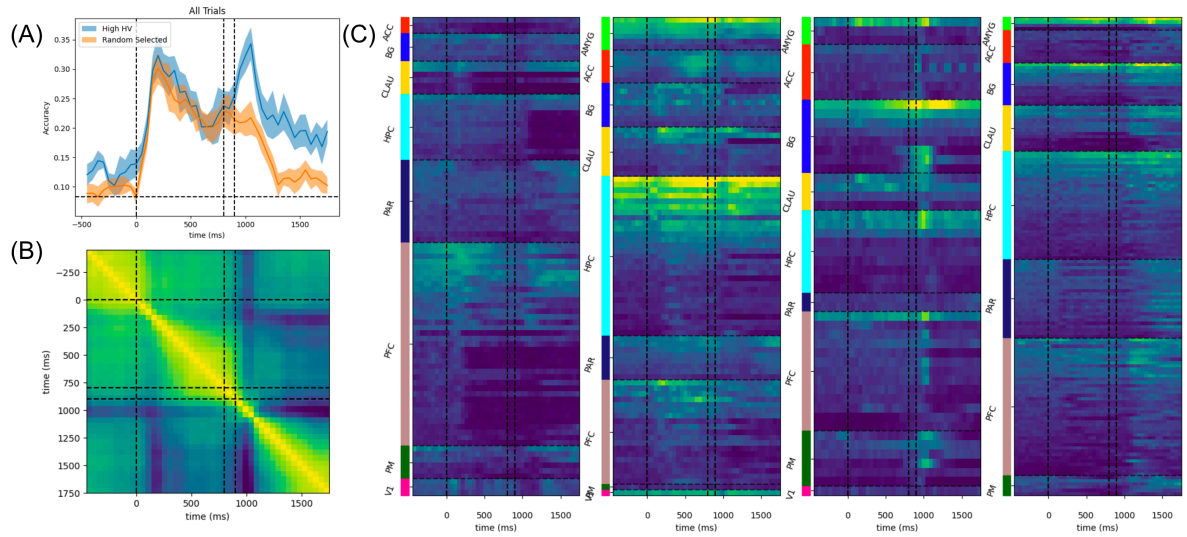

Figure 5.5: (A) Decoding the most likely belief feature (blue) across all sessions, neurons, and timepoints against the control (orange) of a randomly selected feature. The horizontal dashed line indicates chance performance of 1/12. (B) Cosine distance of the mean of the absolute value of the decoding coefficients compared between all timepoints. The square like nature indicates that the coefficients are similar across time, with a few change points. The color scale ranges from 1 to about 0.86. (C) All neurons are grouped into one of four time regions, defined from the "blocks" of (B), based on which time region has the largest mean coefficient. The values are then sorted by maximum mean coefficient. A neuron may decode information across all regions, even though it's sorted into only one region. Brain areas are shown in Fig. 5.4(C). Neurons are thresholded such that at least one value is greater than 0.001. Time regions are defined from -450 to 150 ms, 200 to 900 ms, 950 to 1000 ms, and 1050 ms to 1750 ms. Dashed lines on the time axes indicate the start/end of object fixation as well as the feedback onset.

5.4 Decoding Belief

To provide further evidence, linear decoding can be used to predict neural encoding of the rule belief a decoding approach. Specifically, is the most likely belief feature decodable? In this section, I show decodability during the feedback period, which comes from neurons prominent in hippocampus, basal ganglia, and prefrontal cortex (PFC).

Every trial, three features are chosen by the subject. I assume that only one of these features responsible for the subjects choice. While not necessarily true, it helps with the decoding since every trial has only one associated feature. The chosen feature with the largest associated hidden value from the behavioral model is the most likely feature to be the subjects rule belief. The decoding task is to find a population of neurons that decodes the rule belief of the subject better than chance, which is one out of twelve features.

This decoding method may find a trivial result based on the visual information of the chosen features, where there may be a representation during object fixation or feedback. There is evidence for a working memory representation in the dPCA analysis (Sec. 3.6)[3]. As a control, the same decoding method is applied where randomly selected chosen features are used as the "belief" of the subject. If

[3]Patrick Zhang has also shown decoding of card features during object fixation and feedback.

the decoding accuracy is the same between the two approaches, the decoding must come from a trivial visual representation or memory of all three features. If instead the decoding accuracy is larger for the true rule beliefs, then the subject is likely focusing on the feature with the largest hidden value.

The pseudo-session approach from Bernardi et al. [34] is used to decode the data across many sessions [4]. This method forms "pseudo-trials," where common underlying representations can emerge from multiple sessions while disregarding noise correlations between simultaneous recordings:

1. Divide all trials for all neurons across all sessions by the most likely rule belief.
2. Divide all trials into a train and test set, designated first to avoid overfitting across pseudo-trials.
3. For each neuron and train or test trial out of the total number pseudo-trials, select the activity from a random trial within the respective group of train or test trials. All neurons recorded will be used in the analysis, and a pseudo-trial may consist of neurons from the same session with different real trials.
4. Train the decoders with the train pseudo trials and test with the test pseudo trials.
5. Repeat all steps for total number of iterations.

I use 200 train trials, 50 test trials, 80/20 train/test split, and 10 iterations of train/test splits. This is done independently for every timepoint.

A linear SVM is used to decode the data with regularization of $C = 10^{-5}$, chosen based on the train/test score for random samples with different regularizations. Decoding accuracy for the true belief distribution and control two approaches are shown in Fig. 5.5(A), where there is a jump in decoding accuracy for the true belief distribution during the feedback period. There is some small decoding before the object is selected as well. Also, there is decoding for both approaches during the object selection period and slightly after feedback begins. Given the jump in decoding above the control, the additional behavioral model information helps decoding accuracy. The main contribution of the most likely belief occurs during the feedback period, while decoding of the visual representation or working memory lasts during the object choice and slightly into the feedback period, implying that the belief feature is important for longer than working memory of the choice.

The same plot is found when only correct or only incorrect trials are used (not shown). During correct trials, the increase of accuracy above the control occurs during feedback AND during the object selection. Decoding decreases below the control during object fixation in incorrect trials. This implies that the belief of the subject is more salient during correct trials compared to incorrect trials, which may be especially evident since correct trials tend to occur late in the rule block compared to incorrect trials.

[4] Thanks to Patrick Zhang for figuring out how to adapt this method to the data.

The associated coefficients may reflect where information is being held in the brain. Is the location of the representation stable across time? For all questions related to coefficients, I take the absolute value and mean across iterations to ensure that each neuron contributes, despite positive or negative weighting. The cosine similarity between the mean coefficients at every time point (Fig. 5.5(B)) shows a block-like structure, indicating that the distribution is consistent in a few time periods. Particularly, the biggest change in block structure occurs immediately after feedback, indicating a fairly different representation compared to all other time periods. The biggest difference is around a cosine similarity of 0.86, where a similarity of 0 indicates complete orthogonality. This may be explained by the fact that the similarity is calculated across ~ 700 neurons, where many of them might be similar across time.

The blocks in Fig. 5.5(B) define four time regions: before the object choice, during the object choice, immediately after feedback, and late after feedback. All neurons are grouped into one of these four time regions based on the timing of the largest mean coefficient; while grouped into only one region, a neuron may provide salient information in other time regions. The mean coefficients are plotted for each time region in Fig. 5.5(C). Before object fixation, the little decodability mainly occurs in PFC. During object fixation, the representation occurs in many places, such as the hippocampus and medial temporal lobe, PFC, claustrum, and anterior cingulate. Immediately after feedback, the representation appears in the basal ganglia, hippocampus and medial temporal lobe, PFC, and the pre-motor region. Late after feedback, the representation occurs in most brain areas, indicating that many brain areas may be necessary for computations related to the feature of interest. This is similar to a computation from Steinmetz et al. [162], where an abstract value was encoded in many brain areas.

The results from this section highlight that the most likely rule belief feature is encoded mainly after feedback and during object fixation during correct trials. There are multiple representations, prominently occurring immediately after and late after feedback. This aligns with the immediate feedback given, which will be shown to have many differences depending on the timing (chapter 6). Overall, this provides further evidence for the behavioral model encoding the subjects rule beliefs.

5.5 Representation of Previous Choice

From the behavioral modeling (Sec. 4.4.2), there is a unique behavior when the subject sees a repeated card compared to when all three chosen features are randomly shuffled. Is there a different neural representation when the card is correctly chosen and repeated?

Trials are divided into three classes: trials where all three features are chosen again, two features are chosen again, or one feature is chosen again. These features are likely related to the reason that the subject chose the card on the second trial. To focus on times when the subject "learned" from previous feedback, both trials are required to be correct. There are three hypotheses: the same representation,

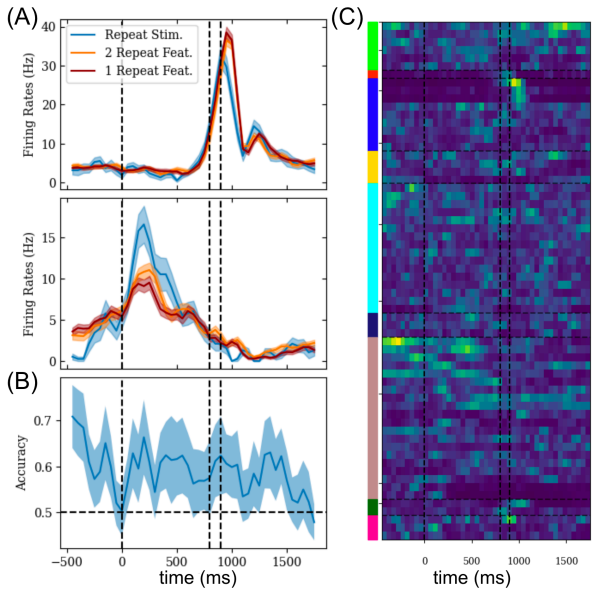

Figure 5.6: (A) Example neural firing rates depending on if the correct card is repeated and chosen again (blue), if two features are repeated (orange), or if one feature is repeated (red). (B) Pseudo-decoding accuracy of a repeated card compared to a single repeated feature. (C) Mean of absolute value of decoder weights (thresholded above 0.01), indicating that decoding ability occurs early in PFC and immediately after feedback in basal ganglia. Brain areas (colors) are shown in Fig. 5.4(C). Dashed lines on the time axes indicate the start/end of object fixation as well as the feedback onset.

regardless of number of repeated features, may occur whenever the subject chooses repeated features. Second, there could be a unique distribution depending on the number of features. Third, and assumed from Sec. 4.4.2, there could be a unique representation for the repeated card compared to choosing 1 or 2 repeated features.

The firing rates for two example neurons (Fig. 5.6(A)) separate based on the number of features shared between the current and previous chosen card. The firing rates are similar when 1 or 2 features are shared but different when the stimulus is repeated, an extremely common phenomenon across many neurons (not shown). Also in these plots, one neuron increases its firing rate when the card is repeated during object selection, which may indicate that the cell responds to repeat cards. The other neuron shown decreases its firing rate for a repeat card after feedback, which may signify that a computation may not be due to the lack of additional information. From these analyses, there are neurons that show differences between a repeated card and non-repeated card, as expected from the behavioral modeling (Sec. 4.3.2). There are also neurons that are different between all three cases (not shown), indicating differences based on number of features.

To compare further, a pseudo-decoding approach (described in Sec. 5.4) is used to decode cases where one feature is re-chosen compared to when the whole card is re-chosen. The decoding accuracy (Fig. 5.6(B)) shows slightly above chance decoding throughout the entire time period, which likely occurs due to many less trials occurring when the subject chooses a repeat card compared to choosing a repeat feature. It's a rare 1 out of 16 event when the previous card repeats. That being said, there

is still evidence for a difference in firing rates, solidifying the idea that representations are unique when choosing a repeated card.

The mean of the absolute value of the decoding coefficients are plotted in Fig. 5.6(C). While there is decoding in many brain areas, prominent areas include the PFC both during object fixation and the feedback period as well as the basal ganglia immediately after feedback. Decoding after feedback may be reflective in computations occurring when only one or two features are chosen again. One could hypothesize that less computations are needed if a whole card is chosen a second time, since no new information occurs. Thus, the coefficients showing locations after feedback are likely reflective of computations of non-repeated stimuli.

From this section, there are firing rate differences when a card is repeated compared to when one or two repeat features are selected. While the decoding approach shows only a minor amount of decoding, examining many neural firing rates show differences in various brain areas. There are neurons that are different only when the card is repeated, and there are neurons that have separate firing rates depending on the number of repeated features. This suggests that both computations may be useful in the brain.

5.6 Discussion

Throughout this chapter, behavioral models have been used to aid the understanding of the brains internal state. Neurons show differences when features with large and small associated hidden values. Using both the GLM analysis and decoding analysis, firing rates are better explained by utilizing behavioral model. Evidence points towards neural differences when the subject chooses a repeated card, as expected from the behavioral model analysis (Sec. 4.4.2). Overall, the neural analysis provides further evidence for the use of the behavioral model in predicting the subjects behavioral state.

That being said, the exact hidden values only weakly correlate to neural firing rates (Sec. 5.1). While this may be due to minimal coding of rule beliefs in the neurons, there may also be a nonlinear population coding of hidden values. As shown, it's useful to explore more general values from the behavioral model, such as the most likely features or small/large probabilities of choice.

While this chapters analyses focused on the object fixation and feedback periods, other time regions may also include information. For example, there may be information related to the subjects belief during the cross fixation. Also, one can be more careful during the feedback period. In the analysis here, hidden values were used from trial t and timed along with the feedback on trial t. However, it's possible that the hidden values update immediately, and neurons may encode hidden values from trial $t+1$. This may be useful in the decoding analysis, where the belief of the rule may rapidly change. One possible analysis is how the representation changes between trials; using the hidden values and choice information may be key to this analysis.

One unanswered question from this chapter is the problem where the GLM analysis implies a belief representation before feedback while the decoding analysis implies a belief representation after feedback. It's possible that this occurs due to differences in correct and incorrect trials, since there is decoding of the belief feature in the correct trials during object fixation. Perhaps the GLM analysis is picking up information from the correct trials. The GLM analysis has the advantage that chosen features and belief features are encoded while the decoding analysis of belief features may be confounded with the chosen features. With that in mind, the GLM analysis might be better at picking up non-stimulus based information.

Another point of interest, the GLM and decoding analysis focused purely on the chosen object fixation. There is also time resolution on when other stimuli are observed. It's possible that similar information can be obtained when focusing on all saccades to new objects. Similarly, perhaps there are differences when the visual features are chosen or not chosen. While somewhat explored in Sec. 7.3.5, analysis on specific features and belief states has not been explored.

Overall, this chapter highlights the use the best fit behavioral model to uncover insights into brain area specific computations related to rule beliefs and choosing repeated cards. The belief of a particular rule is important for many brain areas, but interestingly the neurons responsible for encoding the belief change between object fixation and early/late feedback; there is a very different response immediately after feedback is given. The timing is quick, where information on the belief state occurs for 200-500 ms. With this in mind, the feedback period is a primary time region for understanding updates to the belief state.

One possible view, refined in Chapter 9, is that there is general encoding of the stimulus along with a representation of a belief during the object selection period of correct trials. The encoding likely occurs during the end of the rule block, since that is where the majority of correct trials occur. Immediately after feedback, a very fast computation occurs related related to the belief. While it's unclear what that computation is, the computation may be related to working memory and rule belief interactions combined with feedback. This would explain the idea of the behavioral model, where information of the belief, stimulus, and feedback are all integrated in one computation. Then, late after feedback, there is lingering information about the belief feature. Perhaps the subject is keeping it in memory for the next trial, or there could be further computations taking place. Either way, a primary future analysis is to focus on the feedback period along with the belief state to understand what changes in the neural representation when the belief state changes.

Chapter 6

Representation of Feedback in LFP

An important aspect of this task is the evaluation of feedback. Given that it's impossible to efficiently solve the WCST without integrating feedback and choice history to find the underlying rule, it's expected for there to be differing responses to positive and negative feedback. Trial history could also affect the current feedback. If, for example, the previous few trials were correct, then negative feedback may be surprising. This chapter focuses on understanding differences in feedback, distributed modes, differences in trial history, and ultimately when and where the integration of positive/negative information is computed.

During correct feedback, the subject, monkey S, receives a food reward while simultaneously hearing beeping noises every 200 ms for 1400 ms. If there is a continuous response for this amount of time or if there is a response in 200 ms intervals, the response is likely directly attributed to the reward. This may lead to a boost in 5 Hz power since neural spikes every 200 ms could result add to a population level power increase. While interesting to observe, a more interesting effect is information regarding the processing or integration of feedback; these responses should occur in a different fashion to the 200 ms reward pulses. While this could occur at any time during feedback, the most efficient time to process choice information is immediately after the first reward pulse since this requires the least amount of time for information to be held in memory.

The timing of feedback has interesting implications, since only positive feedback is directly given to the subject. To process negative feedback, or the absence of a positive feedback, the subject would need to realize that no reward was given. It's plausible that a response to negative feedback occurs after a short delay, implying that the subject reliably tracks the timing of feedback. Any response to negative feedback is of interest, since there is no sensory signal for the subject to react to.

There is a growing body of literature studying responses to feedback in various areas of the brain. In a longitudinal study using fMRI on children from ages 8 to 25 [163], they find a plethora of positive and negative feedback differences in the dorsolateral PFC, superior parietal cortex (SPC), supplementary

motor area (SMA), caudate, and anterior cingulate (ACC). They also found that negative feedback is harder to parse than positive feedback. Another longitudinal study also found feedback differences and working memory differences in the dorsolateral PFC, SMA, ACC, and SPC [164]. Others have observed coding of reward prediction error (RPE) as well as bidirectional communication in the dorsomedial PFC and insula [64]. Another study [165], where the subject estimated an unknown target time, has found many differences between correct and incorrect feedback in the medial frontal cortex and PFC. They also show predictive feedback signals for future rewards, indicating adjustment to the task. Bouchacourt et al. [83] found a functional dissociation between rule switching and rule learning based on feedback, both of which may occur in the PFC. Another study explores Hippocampus (HPC) and Pre-Frontal Cortex (PFC) connectivity while learning paired associations [8]. They find that the PFC correlates better with behavioral performance, while HPC reflected feedback. Interestingly, during incorrect trials, they found a causal influence from PFC to HPC from 2-6 Hz. During correct trials in the 9-16 Hz band, they found a causal influence from HPC to PFC. They hypothesize that associative learning may occur in PFC while HPC may guide neocortical plasticity.

While previous work has shown interesting feedback differences in various tasks across many different areas of the brain, feedback is typically assessed in tasks where the feedback either updates the current belief or can directly be used to find the best choice. In the abstract decision making task here, feedback and previous history need to be integrated to isolate the best choice. In this chapter, I investigate different responses to correct and incorrect feedback at various frequency bands in the LFP. There are many, frequency dependent, low-dimensional modes that vary based on feedback and trial history. By examining the timings, locations, and patterns, I observe how feedback is used to update beliefs of the rule.

6.1 Differences in Feedback Responses

The simplest question is, are there response differences between correct and incorrect feedback? I examine the power and channel phase differences in each frequency band. To examine the phase difference, individual channels are used instead of bipoles since bipole subtracting naturally adds random additional phases to every signal. That is, if two signals are subtracted with the same frequency, the resultant signal will be phase shifted. For consistency, individual channels are used for the power analysis as well.

The LFP data from session 2018-08-02 are pre-processed by z-scoring, notch filtering the 60 Hz harmonics, bandpassing to the desired frequency range, rejecting artifacts with ICA (Sec. 2.4), and taking the Hilbert transform to get the amplitude and phase. The frequency bands are 4-5, 10-13, 20-24, 27-37, 65-87, and 120-148 Hz, defined in Sec. 2.7. Raw data is also included. Trials are analyzed around the choice fixation and feedback period. The data are normalized again by z-scoring to remove

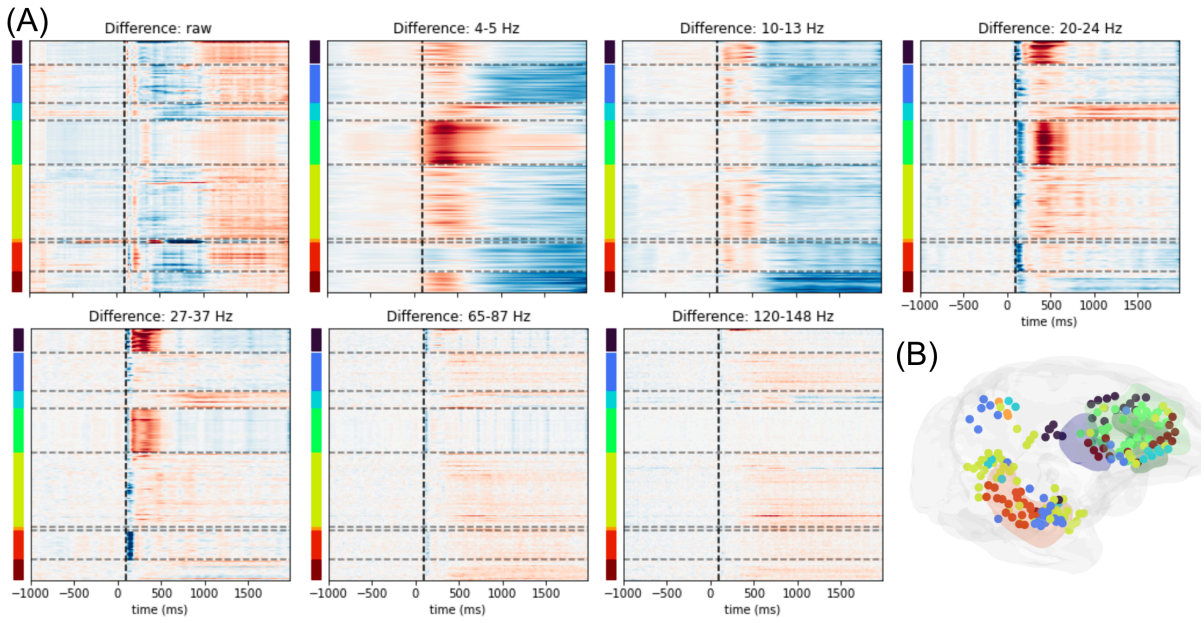

Figure 6.1: (A) Average feedback differences for each frequency band. The color is blue when the signal is larger for correct trials and red for incorrect trials. The signal is concatenated for each channel across bands and clustered into 8 clusters with k-means. Horizontal dashed lines separate the clusters. Vertical dashed line indicates the first reward pulse during correct trials. (B) Locations of each channel. Cluster colors align between (A) and (B).

any significant differences between the power in each band. This is necessary since the raw data is on a different scale compared to the Hilbert power after pre-processing. After pre-processing, the signal is averaged across correct and incorrect trials for each band. The average difference is calculated for each channel and frequency band by subtracting the incorrect mean from the correct mean. To organize, the average difference is concatenated across all frequency bands for each channel and clustered with k-means with 8 clusters. Different numbers were explored, where 8 clusters sufficiently separates common trends without over-dividing the data. This clustering is used for the remainder of this section.

The differences between correct and incorrect trials are shown in Fig. 6.1(A) for each frequency band, and the locations are plotted in Fig. 6.1(B). There are many effects that occur immediately after feedback. Particularly, in the prefrontal cortex (green/black), there is an increase in activity in incorrect trials compared to correct trials in the 4-5 and 20-37 Hz bands. There is also a very quick signal that occurs immediately after feedback during correct trials in both the prefrontal cortex and the hippocampus. Interestingly, the increase in incorrect activity does not happen in the hippocampus. This may indicate different processes that occur in the hippocampus and prefrontal cortex. There are also interesting timing differences after feedback. Between 4 and 13 Hz, there is a switch in signal at $\sim$ 500 ms. It is unclear why the signal spontaneously switches at this point, but it's possible for some computation to end and a new one to begin. Lastly, while small, in the 65-87 Hz band, there are minor increases in activity in the prefrontal cortex every 200 ms or so. This most likely relates to the reward pulses.

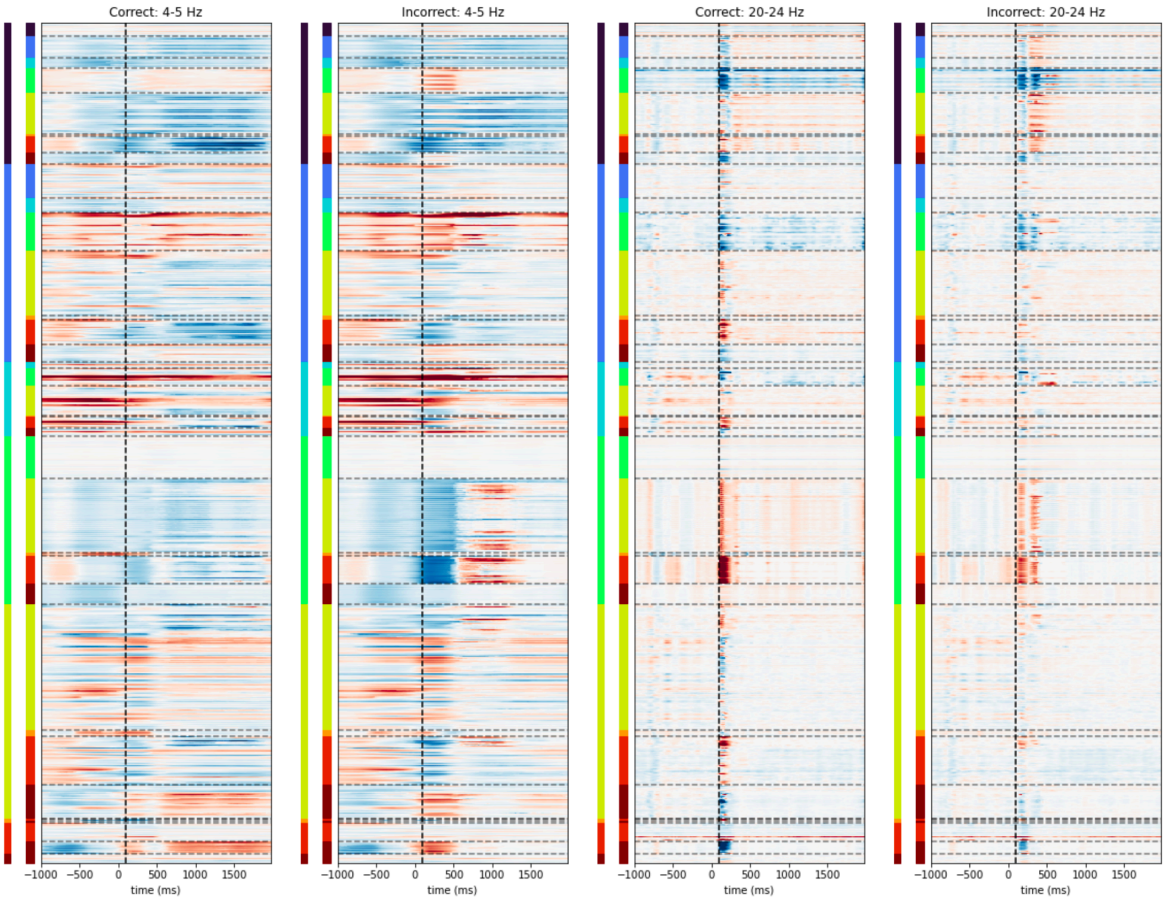

Figure 6.2: Average phase vector differences (Sec. 2.6) between each pair of channels during correct (1st and 3rd panels) and incorrect (2nd and 4th panels) trials within the 4-5 (1st and 2nd panels) and 20-24 (3rd and 4th panels) Hz bands. Phase differences are set to 0 (white) if the average phase vector amplitude is less than 0.11 (correct trials) or 0.125 (incorrect trials), corresponding to a p-value of $\sim 10^{-5}$. Locations are color-coded according to Fig. 6.1(B). In particular, green/black are in the PFC, and red/yellow are near the hippocampus/medial temporal lobe. Channels in the first column are LEADING channels in the second column IF the color is blue.

Overall, the trends in Fig. 6.1 highlight interesting characteristics of the feedback. There are differences between correct and incorrect trials in the hippocampus and prefrontal cortex. Given that the difference in activity occurs immediately after feedback, it is unlikely to be based purely on reward. Further, activity increases for incorrect trials cannot trivially correlate to a simple reward response. Instead, negative feedback responses likely originate based on computation occurring in the brain. Further, new computations may begin after ~ 500 ms in the brain.

I now focus on the phase of the Hilbert transform, which can lead to measures of information flow between brain areas. The data is pre-processed in the same way as before, where the angle of the Hilbert transform is used instead of the absolute value. Channels are grouped into the same clusters as before, indicated in Fig. 6.1(B). The phase is subtracted between every channel pair for each trial and snapshot in time. The phase is averaged, and only average vector differences with a p-value smaller than $\sim 10^{-5}$ are analyzed.

The average phase vector difference (Sec. 2.6) is plotted in Fig. 6.2 for two frequency bands of interest. There are numerous, statistically significant, phenomena of interest. First, the phase differences between the Hippocampus (yellow/red) and PFC (green) during incorrect trials at 4-5 Hz show that, for the first few hundred milliseconds after feedback, the PFC is leading the Hippocampus. Then, the direction switches, and the Hippocampus is leading the PFC. This implies that information is flowing towards the Hippocampus at the beginning of feedback and towards the PFC later on.

Some phase differences occur with stronger amplitudes during incorrect trials compared to correct trials in the 4-5 Hz band immediately after feedback (Fig. 6.2). There is flow between pre-frontal areas, particularly from the PFC (green) towards Frontal Eye Field (FEF, black) areas. There is flow from FEF (black) towards the Hippocampus (red), between Hippocampal areas (red/yellow), and towards the Hippocampus (yellow/red) from the PFC (dark-red). Like before in the amplitudes, there appears to be a switch in activity at ~ 500 ms, indicating a switch in computations.

In the 20-24 Hz band (Fig. 6.2), there are numerous differences between the correct and incorrect trials. Particularly, in some areas, there are two bumps in the phase difference during incorrect trials. This occurs during flow towards the PFC (green) from FEF (black) and towards the PFC (green) from the Hippocampus (red/yellow/blue). There are also single bump activity patterns that occur during both correct and incorrect trials between the hippocampus and PFC (yellow/red/dark-red). The timing is much shorter in this band compared to the 4-5 Hz band, likely due to the higher frequency.

The directions of flow are typically opposite in the 4-5 Hz and 20-24 Hz bands. While flow is towards the Hippocampus (red/yellow) in the 4-5 Hz band, it is towards the PFC (green) in the 20-24 Hz band. This may be indicative of communication through coherence [39], where lower frequencies from higher-order areas may modulate information flow at higher frequencies.

Other frequency bands show similar effects. The 27-37 Hz band has similar effects to the 20-24 Hz band. The 10-13 Hz band is noisier compared to the 4-5 Hz and gamma bands, and the information flow is opposite compared to the 4-5 Hz band within the Hippocampus (red/yellow) and PFC (dark-red); the phase indicates flow towards the PFC. This may also indicate communication through coherence [39]. Phase differences are only calculated up until 37 Hz. Afterwards, the timing is quicker and periodicity effects, while unlikely, may occur.

6.2 Dimensionality Reduced Feedback Differences

Given that the previous section demonstrated large amounts of spatial consistency, it's desirable to reduce the dimensionality and look for common modes across brain areas. Here, I apply Independent Component Analysis (ICA) [126], which finds statistically "independent" modes in the data. Instead of explaining variance like Principle Component Analysis, ICA seeks to separate signals that have unique

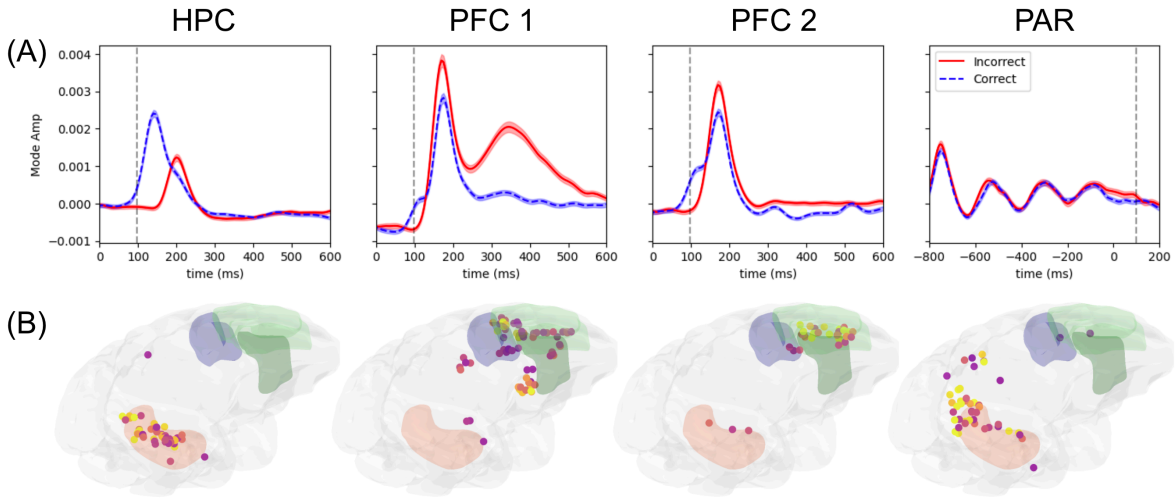

Figure 6.3: Common ICA feedback modes for the 27-35 Hz band. The transformed data is grouped together across sessions based on the average feedback activity. (A) Zoomed in examples from the four most prominent groups, showing the mean and standard error of the mean. There are distinct amplitudes and timings for correct and incorrect trials in the first three cases. The fourth case shows a pre-feedback oscillatory signal. (B) Thresholded (greater than the mean plus standard deviation of all weights) weights for all sessions, typically aligning in the Hippocampus (red) or PFC (green).

origins. During the feedback period, one may assume that feedback signals occur independently of all other brain-wide signals. With ICA, there are better signal-to-noise estimates that showcase interesting feedback related differences.

To pre-process the data, the LFP is bipole subtracted, normalized, notch-filtered around the 60 Hz harmonics, bandpassed to each frequency band defined in Sec. 2.7, and the absolute value is taken. The timing is shortened to -800 to 1900 ms around the end of the choice fixation, averaging over 5 ms bins. The first 42 sessions before the shortened time-out period are used. This leaves the data in a shape of bipoles by trials by time. To apply ICA, the trials and time dimensions are concatenated together. After transforming the data with ICA, the data is un-concatenated back into trials and time, leaving the data in the shape of number of modes by trials by time. Lastly, the dimensionality reduced data is averaged across different feedback conditions. For each mode, I force the sign of the average ICA component to be positive.

Choosing the best number of ICA modes is difficult, since the number of signal "sources" is unknown. While increasing the number of ICA modes from 1 to 10 for a single session, I examined the average feedback dependent modes. I settled on 8 modes as a consistent number to find a few feedback dependent modes.

Common ICA modes occur across sessions, which are grouped together by eye. The most common modes for the 27-35 Hz band (Fig. 6.3) showcase feedback differences in different brain areas. There are two feedback dependent modes in the PFC and one feedback dependent mode in the Hippocampus. There is also an oscillatory mode near the Hippocampus and parietal lobe during the object fixation

period, aligning with the eye movement activity that typically occurs every 200 ms (Sec. 3.3.1). In both PFC modes, there is a boost in power at about 100 ms after feedback begins, and there is a larger boost during incorrect trials. In one of the PFC modes, there is a second boost in power shortly after incorrect feedback. In the Hippocampus, correct feedback leads to a response in 20-30 ms while incorrect feedback leads to a response in over 100 ms, which surround the first bump in activity in the PFC modes. The timing difference may occur since rewards are only given during correct trials and the subject needs to infer lack of reward during incorrect trials. The discrepancy in timing between the Hippocampal and PFC modes may indicate possible feedback related computations that occur at different times.

I now summarize findings from the other bands (not shown), where timings are all approximated. In the 4-5 Hz band, there is a widespread increase of activity in correct and sometimes incorrect trials at 500 ms. In the 9-12 Hz and 13-17 Hz bands, there is a widespread dip of activity in correct and incorrect trials at 300-600 ms. There is also an increase of activity in correct trials at 600 ms and incorrect trials (less often) at 700 ms. The 20-24 Hz band has the same effect as the 27-35 Hz band with less defined features. In the 45-55 Hz band, it is common to see a peak in activity in the PFC at 150 ms for both correct and incorrect trials. There is a second peak at 300 ms for incorrect trials. In the 65-87 band in the PFC, there is activity at 100 ms during correct trials and 150-200 ms during incorrect trials. Groups of bipoles, active for all trials at 150-200 ms, split into two groups where activity increases in either correct or incorrect trials at 300 or 400 ms respectively. In the 120-148 band in the PFC, there is activity at 100 ms after correct feedback and activity at 150-200 ms after incorrect feedback. Also in the PFC, there are some bipoles with large peaks in activity at 400 ms. From 20-55 Hz, there are sometimes large increases in activity during incorrect trials from 500 to 1500 ms. Lastly, from 9-35 Hz, there are modes with activity every 400 ms (every other food pulse/beep). From 45-148, there are modes with activity every 200 ms (every food pulse/beep).

Throughout all bands, there are activity patterns that differ between correct and incorrect feedback. Responses typically occur immediately after feedback, but sometimes align with food pulses. Given the spatial organization and timing differences, it's possible to understand possible computations and brain locations. I'll explore possible interpretations later in this chapter.

6.3 Differences based on Previous Feedback

To put the feedback differences into context, I split the modes by the previous trials feedback. If the previous and current trials are correct, which typically occurs during the end of each rule block, then the subject likely has knowledge of the current rule. If the previous trial was incorrect and the current trial is correct, then the subject likely gained new information. If the previous trial was correct and the current trial is incorrect, depending on the number of previous correct trials, then the subject may be

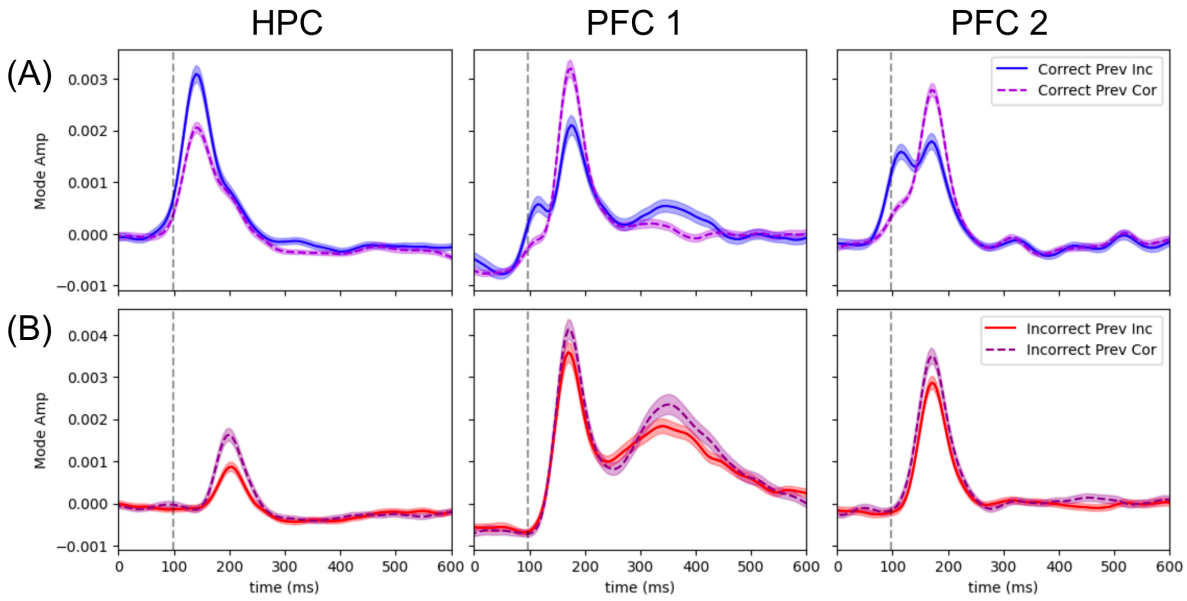

Figure 6.4: Separation of correct (A) and incorrect (B) feedback modes based on the previous feedback. Titles and mode locations align with Fig. 6.3(A).

surprised that they are wrong. Lastly, if the previous and current trials are incorrect, the subject is likely unsurprised and does know the rule.

Differences in amplitudes for previous correct and incorrect feedback (Fig. 6.4) occur in a similar fashion across sessions. In the Hippocampus, the amplitude is increased when the previous trial's feedback is opposite to the current trial's feedback. That is, when surprised, the amplitude increases. In the PFC, the amplitude increases when the previous trial was correct in all cases. Also, when the previous trial was incorrect and the current trial was correct, there is an increase in amplitude earlier than the activity increase in the Hippocampus. This activity increase also occurs in the PFC in Fig. 6.3, and the previous feedback amplifies the effect.

Here, I highlight findings from other frequency bands (not shown). In the 4-5 Hz band in the Hippocampus, there is typically a large correct response after an incorrect trial around 800 ms after feedback. In the 9-12 Hz and 13-17 Hz bands in the PFC, there is an immediate large correct response that amplifies further when the previous trial was incorrect. From 9-12 Hz, there is a HPC mode that increases it's incorrect feedback power at 1000 ms onward when the previous trial was incorrect. In the 20-24 Hz band, there is a Hippocampal mode that increases it's power when the previous feedback misaligns with the current feedback. This occurs immediately after feedback, and it occurs slightly earlier for correct trials. There is a PFC mode at around 400 ms that increases it's incorrect response power further when the previous trial was incorrect. Also in the PFC, there is an immediate correct response that occurs earlier by ~ 20 ms when the previous trial was incorrect. In bands from 45 Hz onward, there is a correct response at 180 ms that heightens when the previous trial was also correct. In these bands, there is a secondary bump, immediately after correct feedback, that occurs when the previous trial was

incorrect. This bump is in general larger at higher frequency ranges. There is a 45-55 Hz band in the HPC, immediately after correct feedback, that has a heightened response when previously incorrect. Also in the HPC, there is a 65-87 Hz response to correct feedback with a long time-scale heightened response when previously incorrect: the response is larger for about half a second after feedback occurs.

6.4 Discussion

The Hippocampus (HPC) and prefrontal cortex (PFC) show differences in feedback responses, which may be necessary in performing this task. The first response occurs in the PFC after correct feedback when the previous trial was incorrect. The second response typically occurs in the HPC, which typically has a larger response when the previous trial was incorrect. Next, PFC modes begin for both correct and incorrect trials, where the response is heightened when the feedback is opposite to the previous feedback. Then, there is a feedback response in the incorrect trials in the HPC, where the response is amplified when the previous trial was correct. All of these responses typically occur shortly after feedback. From Fig. 6.2, there is communication between the HPC and PFC. In the 4-5 Hz band, information flows towards the HPC; this effect is stronger during incorrect trials. In the 20-24 Hz band, information flows towards the PFC; this effect is stronger during correct trials. These effects agree with Brincat et al. [8], although Brincat et al. found the high frequency effect in the 9-16 Hz band. Also, Brincat et al. used a red error screen to convey negative information, different to this task where negative information is not directly conveyed; this suggests that negative information processing is more abstract than a pure sensory representation. After 500 ms, there are responses in lower frequency ranges. Particularly, there are amplified responses when the previous trials were incorrect from 4-12 Hz in the HPC. Further, there is a flow of information at 4-5 Hz in incorrect trials towards the PFC.

Brincat et al. [8] hypothesize that associative learning may occur in the PFC while the HPC may guide neocortical plasticity, a hypothesis relevant to this task. Given the timing related response in the HPC, the HPC may be tuned directly towards feedback. The HPC response is also amplified when "surprised," or when the feedback is opposite to the previous trial. At the same time, there is information flow towards the PFC from 20-24 Hz, which may be needed for the PFC to integrate feedback with current knowledge to update the belief.

Several questions remain. First, why is there low frequency information flow from the PFC towards the HPC immediately after feedback? This may be best explained with the communication via coherence hypothesis [39], where low frequency communication may help the brain attend towards information coming from the HPC. Second, like the HPC, the PFC has an early response to correct feedback when the previous trial was incorrect. This effect may also be due to information flow from the HPC. Or, other pathways to the PFC may exist. If the PFC is responsible for associative learning, then receiving correct

feedback when uncertain is particularly important. Third, note the large response in the PFC when the previous trial was correct and the current trial is correct (Fig. 6.4). The large response difference might be enough to ”confirm” the currently held belief. In the same vain, ”surprising” incorrect feedback, or when the previous trial was correct, may lead to different computations when updating the belief. Finally, there is the secondary information flow after 500 ms towards the PFC in low frequencies (4-5 Hz), which may indicate further computations between the two areas.

Overall, the feedback responses agree with results from Brincat et al. [8]. There are feedback and previous feedback specific computations, which occur immediately after feedback. While not explored in this chapter, other information may also be found in this time period. Reward prediction error is closely related to the previous feedback metric, and it may be better for understanding computations. In general, the behavioral model (Chapter 4) may provide better metrics for surprise than previous feedback.

Chapter 7

Additional Data Analysis

This chapter focuses on an early unsuccessful attempt at rule decoding, phase reset, and eye movement correlates. While not quite fitting into previous chapters, the results here are useful in understanding particular neural correlates.

7.1 Rule Decoding

To find the underlying mechanics of learning, it's desirable to find a representation of the rule, or feature that the subject is focusing on. While explored in the context of behavioral models in Chapter 5, this section focuses on the unsuccessful initial rule representation search with decoding methods.

Before diving into the decoding analysis, it's important to understand the scale and difficulty of the problem. To find a rule representation, the subject needs to think in terms of features. Based on Chapters 3 and 4, a feature based representation is likely. Then, a simple idea is that rule representations occur during the last few consecutively correct trials. Although, one doesn't know when the rule representation begins. Further, analyses from Chapter 3 show that the subject thinks of multiple features during the last 16 out of 20 trials. However, one can argue that, subject to noise, these trials could still reveal a representation of the current rule.

Focusing on single sessions has drawbacks. Due to the randomness of the rule generation, some sessions use only 11 out of the 12 possible rules. Sometimes a single rule will occur once. On average, after pre-processing, there are 28 blocks per session (14 for the shortest and 53 for the longest). For 28 blocks and 8 trials per block (assuming the last 8 trials are when the subject is thinking of the current rule), there are on average 224 trials to use for decoding 12 rules. If the rules were equally represented, then 18 trials can be used to distinguish each rule representation; there are less trials when the dataset is balanced. This ratio may lead to overfitting. Although, one could improve this problem by focusing on a subset of features. Further, overfitting is a problem when considering the number of channels. Per

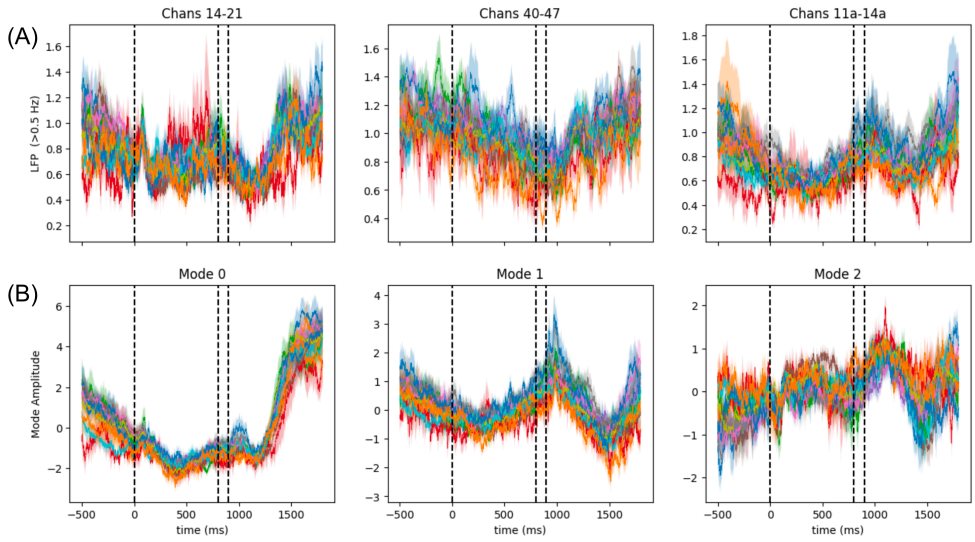

Figure 7.1: (A) Example LFP traces for different bipoles during session 2018-09-18. The LFP is high-pass filtered above 0.5 Hz, normalized, and notch filtered around the 60 Hz harmonics. (B) The top 3 PCA mode amplitudes. Each trace in (A) and (B) is the average activity/amplitude per rule during the correct criterion trials.

session, there are $\sim$ 200 LFP channels, $\sim$ 70 LFP bipoles, and $\sim$ 35 single units. At best, with units, there would be 18 trials to distinguish 12 rules with a 35 dimensional vector. Extended measures, such as bootstrapping, are needed to avoid overfitting.

Timing is also important. If focusing on the feedback period, then there are large overlapping responses to reward. If focusing on the object search period, there is the overlapping visual information of the stimuli; one needs to distinguish between belief representations and visual signals. Perhaps the cross fixation period is best. However, to find a rule representation, the subject needs to be focusing on the feature during this time.

Lastly, I focus on the LFP here, which may not easily find rule representations. While individual spikes have shown selectivity to rule like features in other tasks [166], it's questionable how likely a rule representation would occur in LFP. One hypothesis is that some population of individual neurons fire when thinking of a particular rule. The LFP, which measures 100's to 100,000's of neurons, may measure multiple populations responsible for different rules. This could lead to indistinguishable signals for the different rules.

No Rule Representation Separation

In the ideal case, there is a unique time-locked representation for every rule. To check, the signal is averaged across all criterion trials per rule, where the criterion trials are the last 8 out of 8 or 16 out of 20 correct trials (Fig. 7.1(A)). While there are minor differences, there are no large signal differences between all 12 rules. To check if a population of bipoles show rule representations, the data is transformed

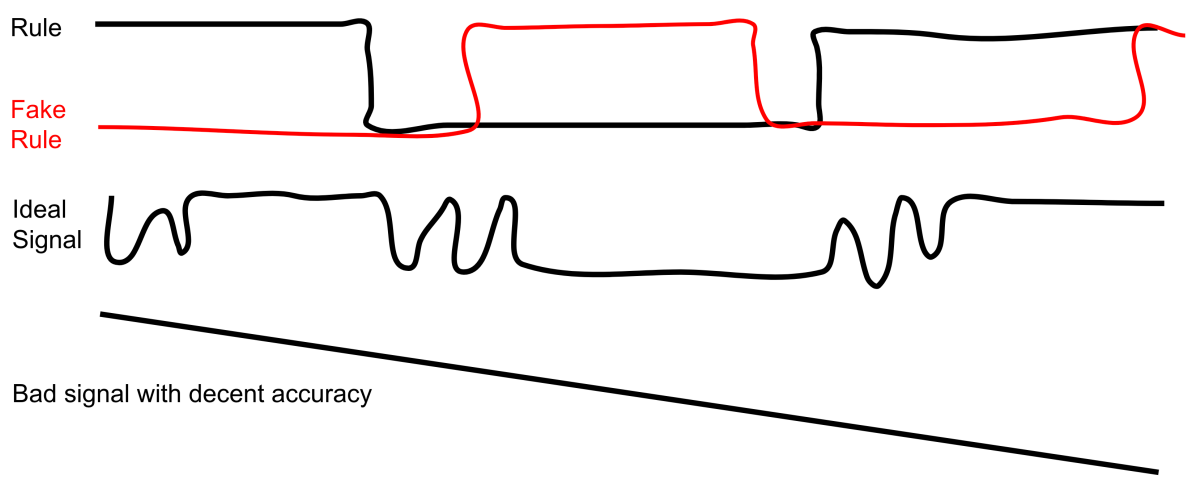

Figure 7.2: Cartoon of the rule decoding problem. Imagine two rules that alternate based on the "Rule" line. An "ideal" signal would alternate between the rules when the subject is unsure, and the signal would converge at the end of the rule block. A "bad signal," which does not align with the rule, could still classify the rule with above chance accuracy. With "fake rules," indicated by the red line, classification accuracy should decrease if the signal is "ideal" while accuracy would remain the same with a "bad signal."

with Principal Component Analysis and then averaged across rules. The principal components (Fig. 7.1(B)) do not show any shifts in rule representation. The same phenomenon occurs for all other bands.

Another way to check for rule representation is to use classification methods to decode the rule from activity. With the assumption that the LFP is constant for some time period in the task, the signal-to-noise ratio is improved by using multiple samples per trial. Train and test sets are split by trials, and consecutive timepoints are set to either the train or test set. To boost the signal-to-noise ratio, the data is bootstrapped by shuffling the train/test data at each timepoint for a particular channel.

Since the rules have a temporal structure throughout each session, trials are not independent and nonsense correlations [141] arise. The problem is highlighted in Fig. 7.2 for two rules. A "bad signal," which slowly modulates across the session, would decode the rule with above chance accuracy. "Fake rules," where the structure of the sequence of rules is retained while shuffling exact timings, can be used to address the slow modulation. An ideal signal's decoding accuracy of the fake rule should decrease while a "bad signal" would retain above chance accuracy. Fake rules are defined based on real rule statistics and occur when the subject is performing the task. Rules switch after a random number of trials, determined by the mean and std of the subjects block length. There is a minimum of 8 trials before the rule can switch. Trial lengths are randomly determined based on the mean and std of the subjects trial length. Timepoints for classification begin at the "trial starts" of the fake rule dataset. Trial lengths are adjusted so that the classification times avoid large responses during the feedback period.

To classify, the bipoles are first normalized and the 60 Hz harmonics are notch filtered. The bipoles are bandpassed, hilbert transformed, and the absolute value is taken. The data is averaged for 100 ms bins, overlapping every 50 ms. Data points are taken from -250 ms to 750 ms around the beginning of

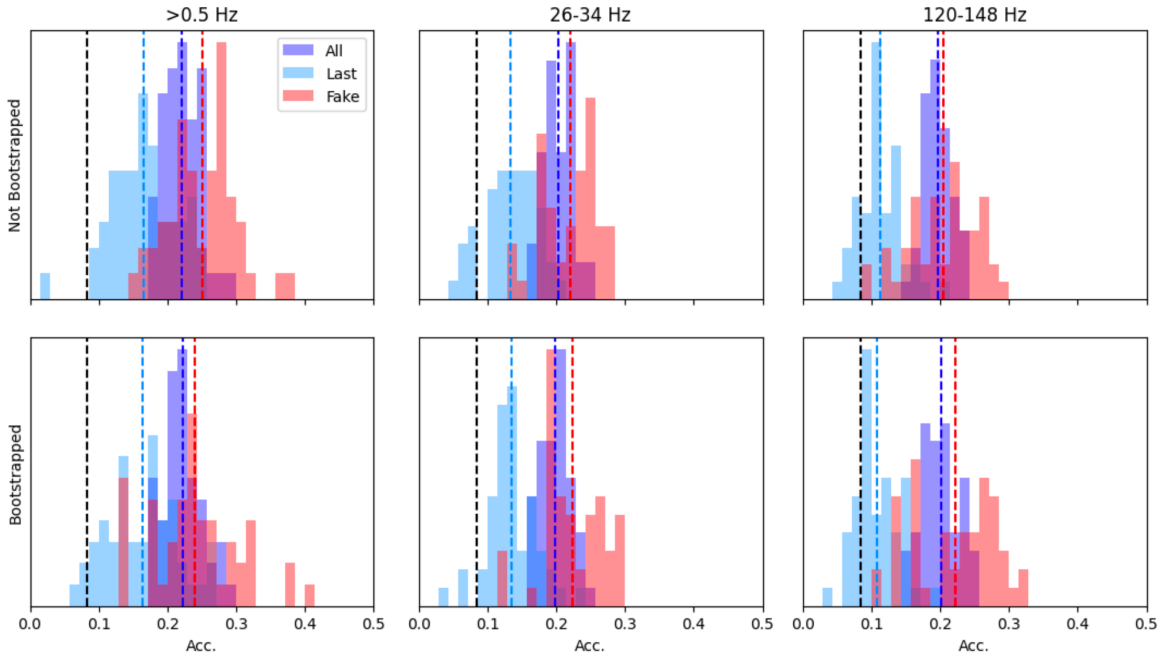

Figure 7.3: Rule classification for different frequency bands (columns) when bootstrapped (bottom) or not bootstrapped (top). 50 iterations of subsampling lead to distributions and means (colored dashed lines) of decoder accuracies for all trials (blue), the last 8 out of 8 or 16 out of 20 correct trials (light blue), and fake rules (red). "Chance" is 1/12 (black dashed line). If rule decoding was present, decoding should be stronger for real trials than the fake rule trials.

the cross fixation. The data is classified based on all trials, the last 8 out of 8 or 16 out of 20 correct trials, non-bootstrapped data, bootstrapped data, on real rules, and on fake rules. The data is 80/20 split into train datasets and test datasets.

The decoding accuracy (Fig. 7.3) is similar between real and fake rule trials. The decoding accuracy decreases for the last 8 out of 8 or 16 out of 20 correct trials. The same effect occurs for all observed frequency bands and sessions. Overall, this method doesn't recover rule representations, despite the above chance decoder accuracies. Most likely, the above chance accuracies are due to slow drifts of the signals across time, resulting in nonsense correlations.

7.2 Phase Reset

Evidence has shown that the phase of the LFP resets during critical times during tasks [167–169] and saccades [43]. Modulation of the phase may be important for setting up plasticity and memory formulation [170–172], particularly in the hippocampus. Here, I check if phase reset occurs in monkeys during the WCST.

To maintain similarity with the 3-12 Hz theta band definition [43] and to ignore saccade related artifacts (Sec. 3.3.1) in the 4-5 Hz band, I focus on the 9 to 12 Hz band. Many [173–175] have argued that "true" phase reset only occurs when the amplitude remains unchanged. Thus, both the amplitude

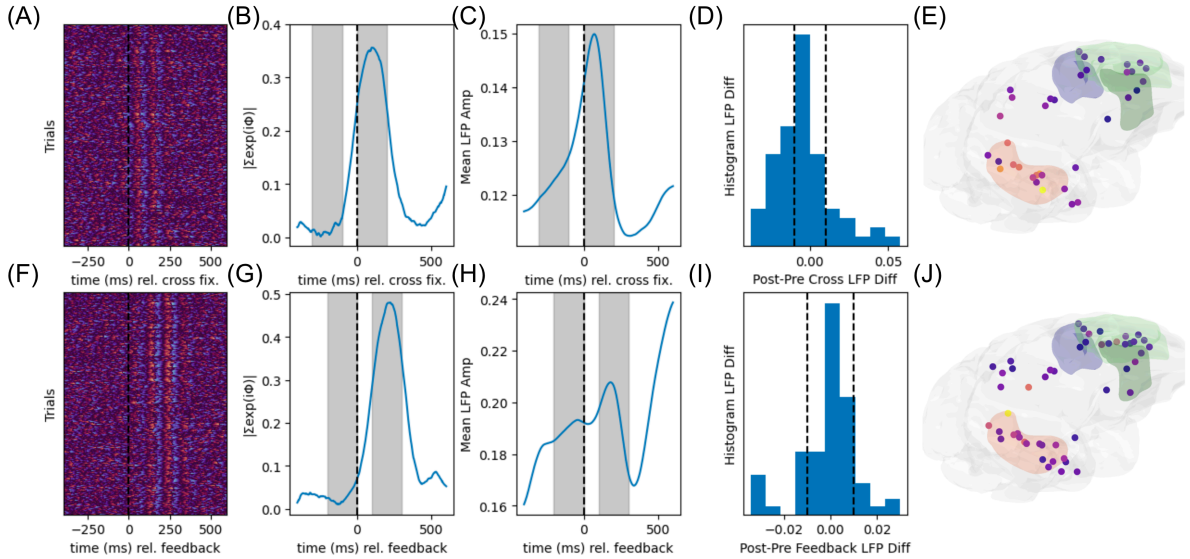

Figure 7.4: Phase reset in the hippocampus during cross fixation (top row) and feedback (bottom row). (A, F) Example bipoles and the theta phase for each trial, showing phase reset after $t = 0$ms. (B, G) Average phase amplitude (Sec. 2.6), showing phase locking after $t = 0$ms. (C, H) Associated LFP average power, where there are sometimes amplitude changes after $t = 0$ms. (D,I) Histogram across bipoles of subtracted pre and post mean LFP amplitudes (gray bands in C and H). (E, J) Bipole locations that meet the LFP difference threshold ($|\text{diff}| < 0.01$, black dotted lines in (D, I)); colors indicate the pre/post phase locked difference (gray bands in B and G), showing differences in the medial temporal lobe and hippocampus.

and phase of the bipoles are examined. All bipoles from Monkey S's session on 2018-09-18 are notch filtered around 60 Hz, normalized, bandpassed from 9 to 12 Hz, and Hilbert transformed. Both the angle and absolute value of the Hilbert transform are used. The angle of example bipoles for the cross fixation and feedback periods (Fig. 7.4(A, F)) show consistent phase reset. To examine further, I use the phase addition metric (Sec. 2.6) to find the average phase amplitude, where a value of 1 indicates perfect phase locking and 0 indicates no phase locking. Per time point, Fig. 7.4(B, G) shows the resultant length, which increases immediately after cross fixation and feedback. The associated mean LFP amplitude per time point (Fig. 7.4(C, H)) shows that there are sometimes large LFP amplitudes during phase locking events.

The pre-post difference (gray bands in Fig. 7.4(C, H)) is calculated for all bipoles Fig. 7.4(D, I), where I arbitrarily set a threshold of ± 0.01 to ensure that the LFP amplitude doesn't vary too much. Finally, the locations of the bipoles that meet the criterion are plotted in Fig. 7.4(E, J), where the colors show the resultant vector difference between the pre and post times (gray bands in Fig. 7.4(B, G)). The largest differences, for both the cross fixation and feedback periods, occur approximately in the medial temporal lobe and hippocampus. This verifies previous results [43, 167–169], where phase reset occurs after critical task events in the hippocampus.

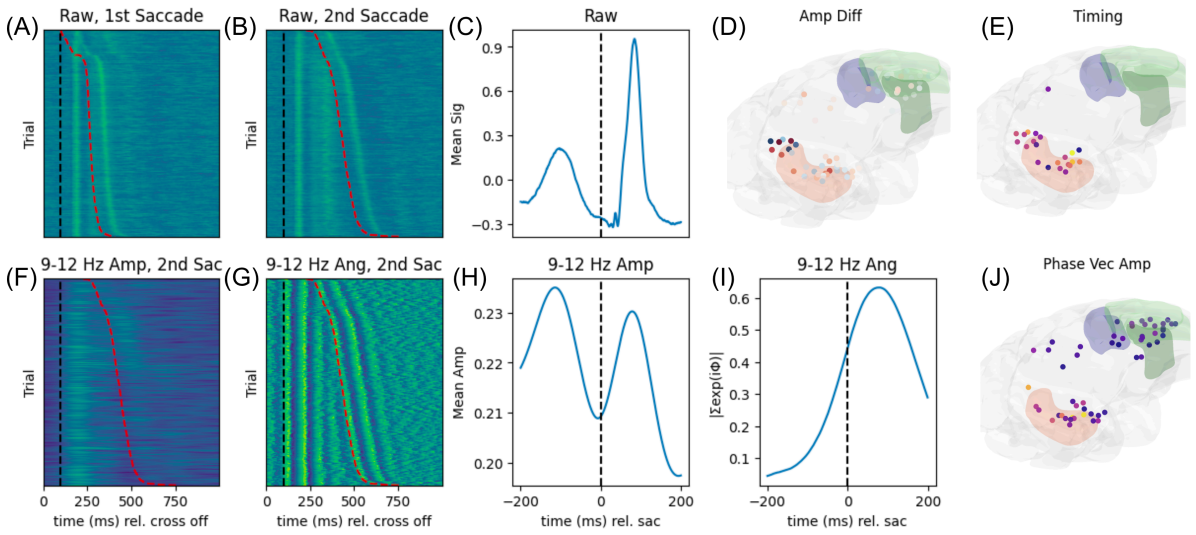

Figure 7.5: (A, B) Example bipole raw LFP after the objects turn on (black dotted line), showing a saccade response; trials are sorted by the first (A) or second (B) saccade. There is also a response after the objects turn on. (C) Averaging the example bipole's raw LFP around the saccade start time, where the raw LFP increases after saccading. The pre-saccade activity is likely due to the previous saccade, typically occurring 200 ms earlier. (D, E) The post saccade response, defined as the timing of the maximum or minimum LFP power, is found for all bipoles in session 2018-08-02. The pre-saccade response is defined as 100 ms prior to this. (D) The amplitude difference between the post and pre-saccade responses, indicating power increases (blue) and decreases (red). (E) Timing of each response, thresholded with a post minus pre amplitude difference > 0.2. Colors range from 0 ms (dark) to 150 ms (yellow). The amplitude (F) and phase (G), sorted by the second saccade, for the same bipole, bandpassed between 9-12 Hz. The phase resets. The mean amplitude (H) and sum of phases on the unit circle (I) for all saccades, indicating phase clustering (I). (J) Bipoles sum of phases (color) and locations, thresholded by the LFP difference between post and pre-saccade LFP 9-12 Hz amplitude (> 0.005), indicating phase clustering in the Hippocampus.

7.3 Eye Movement Correlates

LFP data may correlate with saccades, or eye movements, where there is evidence for LFP evoked responses [176], theta reset [43], and even decoding of movement intention [177]. During the object selection period, eye movements are the most reliable timings for when the subject first observes each object, which may lead to evoked responses or even complicated ideas such as choice intention.

7.3.1 Evoked Responses and Phase Reset

I check for evoked responses in the raw LFP for each bipole in session 2018-08-02, focusing on the object search period. AN example bipoles signal is shown in Fig. 7.5(A,B), where there is an evoked response shortly after each saccade. There is also a large response shortly after the objects appear on the screen, likely a response to the visual stimuli. To analyze all bipoles simultaneously, the raw LFP is averaged across all saccades (Fig. 7.5(C)), where there is an evoked response. There is also a sizeable response approximately 200 ms earlier, which occurs since eye movements occur every 200 ms (Sec. 3.3.1).

To find the size and direction of the evoked response, the maximum absolute value of the average

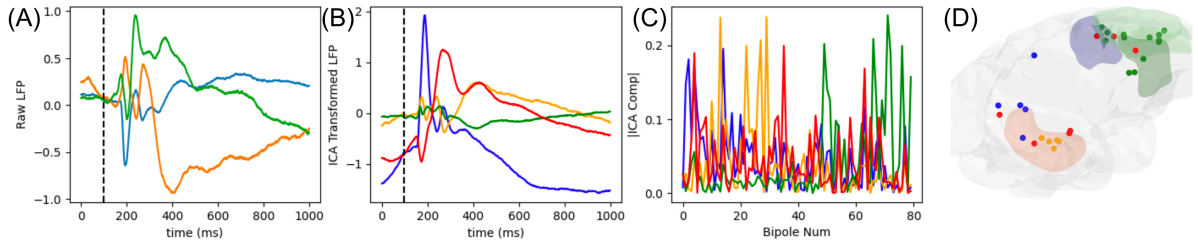

Figure 7.6: (A) Example mean bipole LFP signals (Monkey S's session 2018-08-02) after the fixation cross turns off (t=0 ms) and the objects turn on (black line). After running ICA with four components on all trials, the mean transformed data (B) and absolute value of the ICA components (C) show different average activity patterns with sparse representations. This is contrary to an artifactual hypothesis, where a single ICA mode dominates the entire brain space. The largest component's amplitude is selected per bipole (D), where the bipoles are thresholded such that the difference between the largest and second largest ICA component is > 0.04. The threshold is decided based on a histogram of differences between the first and second largest ICA component. The signals are spatially clustered, with the exception of the red mode. Colors are aligned in (B-D).

signal is calculated for each bipole. In Fig. 7.5(C), this would be approximately 0.9 at a timing of 70-80 ms. I define this as the post saccade response. Then, since eye movements occur every 200 ms, the pre saccade response is defined to occur exactly 100 ms before the post saccade response. Bipoles with no evoked response are excluded. The difference between the post and pre saccade response are calculated (Fig. 7.5(D)), where both positive and negative evoked responses occur at the tip of the medial temporal lobe/hippocampus. The exact timing (Fig. 7.5(E)) fluctuates from 0 to 150 ms after the saccade.

Next, to investigate phase reset during saccades [43], the LFP is bandpassed from 9-12 Hz (theta band). The amplitude (Fig. 7.5(F)) and phase (Fig. 7.5(G)) are shown for the example bipole, where there is a minor increase in theta power and phase locking to each saccade. Similar to the raw data, the average theta amplitude is found surrounding times to each saccade in the object search period. (Fig. 7.5(H)). Again, the power increases approximately every 200 ms, aligned with the 200 ms saccade rate. To examine the phase clustering, the phase is placed on a unit circle and summed together $1/n|\sum_j \exp(i\phi_j)|$. The larger this statistic, the more the phase is clustered. As shown in Fig. 7.5(I), the phase is locked at the post saccade response time. Similar to the phase resetting analysis in Sec. 7.2, I threshold bipoles according to the difference in mean LFP amplitude at the post and pre saccade response times ($|\text{diff}| > 0.005$). The value of 0.005 is chosen based on a histogram of LFP amplitude differences (not shown). Thresholded bipoles (Fig. 7.5(J)), similar to Sec. 7.2, show phase resetting in the medial temporal lobe/hippocampus.

7.3.2 Response to Object Onset

In 7.5(A,B,F,G), there is a response in activity shortly after the objects appear on the screen. One question is whether this is artifactual. If it truly was an artifact, this signal would be present in all bipoles at the exact same time. In three example average bipole activities (Fig. 7.6(A)), there are different patterns, indicating that this is not likely an artifact.

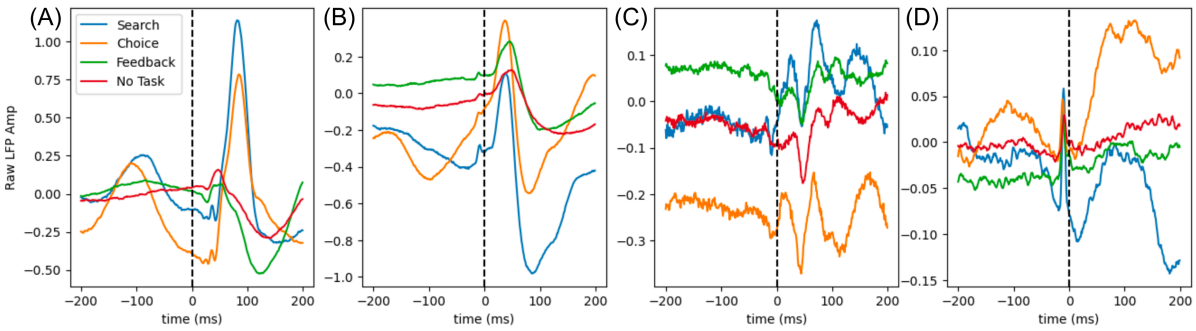

Figure 7.7: Example raw LFP bipole activities from session 2018-08-02, alied to the beginning of each saccade (t=0 ms) from the search period, fixation on the choice object, feedback period, and periods outside when the task was run. The signal is averaged over all saccades. Activity differs between periods, likely due to background activities. In (A,B), the traces follow trends between the search/choice period and the feedback/no task periods, indicating different neural mechanisms when choices are made compared to purely saccade related activity. In (C), there are similar activities across traces, indicating purely saccade related activity. In (D), there are mixtures of similar activity right before the saccade and different traces after the saccade occurs.

To check further, Independent Component Analysis (ICA) can be used to find independent signals in the data. If artifactual, one mode should capture signals in the entire brain. The data is first shaped into a number of bipoles by a concatenated array of time ($98 < t < 298$ms) for all trials. I used four ICA components as a middle ground between over explaining the data and assuming enough modes to explain the data well enough. The LFP data is transformed into the ICA subspace across the entire time series, averaged across trials (Fig. 7.6(B)); there is unique activity for each mode. Further, the ICA components (Fig. 7.6(C)), show sparse representations of each mode across the brain. If there was an artifact, then there should be a mode that dominates the variance across all bipoles. Instead, modes are dispersed.

Given the sparse representation in Fig. 7.6(C), one may ask if the activity patterns in Fig. 7.6(B) are brain area specific. To check, the largest and second largest ICA component (Fig. 7.6(C)) are found per bipole, thresholded by a difference of 0.04 between the largest and second largest ICA component. The largest mode per location (Fig. 7.6(D)) shows spatial clustering, with the exception of the red mode, which is spatially dispersed.

In conclusion, there are unique neural computation per brain area after the objects turn on. This may initialize the timing for the brain to start searching for a card to select. Further, there may be different responses based on which stimuli appear. This forms a testing ground for behavioral correlates time locking to object appearance.

7.3.3 Responses By Task Interval

During the object search interval, there are many brain-wide computations and saccades typically occur every 200 ms. There are other intervals with different characteristics. For example, when receiving feedback, saccades do not occur every 200 ms, but there is still likely to be brain-wide computations.

In this section, I focus on the structure of the LFP for different intervals to find unique properties. I neglect the cross fixation period, which is subject to large preparatory background activity patterns.

I focus on four specific time intervals. First, the object search period can be divided into two intervals: before and after a choice. Before the choice, there are large, inter-object saccades. Afterwards, there are micro-saccades within the chosen object. Both have visual information and 200 ms inter-saccade times. Then, in the feedback period, saccade timings are sporadic and there is no visual information. Also, when the task is not performed, there are no computations and no visual information. With this in mind, the feedback and no-task intervals are likely driven by pure saccade information while the search period saccades are driven by visual information and interpretation.

Fig. 7.7 shows four example bipoles, where the raw LFP is averaged across all saccades for the four aforementioned time intervals. In Fig. 7.7(A,B), activity patterns are similar between the search and choice periods and similar between the feedback and no-task intervals. However, the activity is dissimilar between the search/choice and feedback/no-task intervals. These bipoles are likely influenced by either visual information or neural computations related to solving the task. In Fig. 7.7(C), the bipole has similar activity between all four time intervals, indicating purely saccade driven activity. Fig. 7.7(D) shows a mix of activity. There is similar activity between all four time intervals immediately before the saccades occur, which is most likely activity that is driving the saccade. Afterwards, the activity differs between the search/choice and feedback/no-task intervals. Across traces, the different means are likely due to other, non-saccade related neural correlates. For example, there are different background activities in the feedback period compared to choosing an object. This highlights the difficulty in systematically comparing all traces for all bipoles.

Overall, there are examples of differences between the search/choice and feedback/no-task time intervals, suggesting that there are interesting neural computations occurring when visual information appears. For example, there could be information on the shape/color/pattern of the object, or even information related to a continuance or stopping of the search. Otherwise, there are also examples of purely saccade driven activity patterns.

7.3.4 Visual Information After Saccading

How does the activity depend on the visual features? I focus on saccades that occur during the search period but before fixating on the final choice. I also ignore any saccade that may be deemed as "stereotypical" (Sec. 2.9), to ignore any possible noisy information. Then, I align all data to the end of the saccade. Lastly, I enforce that all saccades have a minimum fixation of 100 ms on the object afterwards, ensuring at least 100 ms of visual processing.

To begin, I define two types of saccades. The first one is when the gaze ends up within the object. For the second one, the gaze must be outside the object but within the defined bounding box. The average

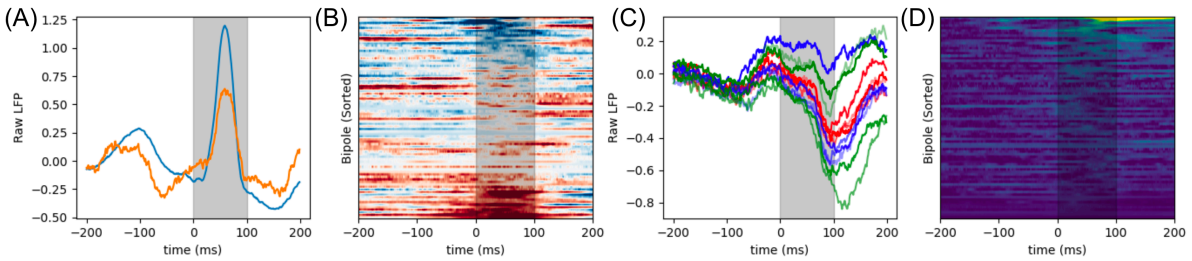

Figure 7.8: Difference in visual information after saccading, relative to the end of the saccade. Only saccades with a minimum 100 ms fixation (grey band) are included. (A) Example bipole's average raw LFP after saccading to an object. The blue trace is when the gaze is within the object. The orange trace is when the gaze is inside the bounding box but outside the object. (B) All bipoles mean difference (within object minus outside object, blue is positive), sorted by the maximum difference within the 100 ms band. There are many differences, indicating that the subject may gather more information when gazing at an object compared to near it. (C) Raw LFP, averaged based on individual features (color=red, shape=blue, pattern=green), for the bipole with the largest difference in signal. Only saccades where the gaze lands within the object are included. This bipole shows significant differences between the 12 features. To calculate the largest difference in signal per bipole, the maximum difference during the fixation period (grey band) is calculated for every pair of features. The two features with the largest difference is used for (D), showing the absolute value of the difference between signals. Only a handful of bipoles show separation, indicating that separation is rare. Similar effects are seen in specific frequency bands (not shown)

LFP for an example bipole (Fig. 7.8(A)), shows a clear difference between the two types of saccades, a result that occurs for all bipoles (Fig. 7.8(B)). There are many bipoles with signal differences between the two types of saccades during the 100 ms fixation period. This indicates that the subject may be receiving different information when actually gazing upon the object instead of being nearby. With this in mind, only saccades of the first type (gaze within object) are analyzed further.

Next, does the LFP encode visual information when saccading to it? The raw LFP is averaged for all saccades with each of the 12 features. Then, I find the maximum difference between the signal for each pair of features during the 100 ms fixation period. Then, the pair of features with the maximum difference is found per bipole. From this, the bipole with the maximum difference (Fig. 7.8(C)) shows differences between each feature. Then, the differences for every bipole (mean subtracted for the pair of features with the largest difference) are shown in Fig. 7.8(D). Only a handful of bipoles have a difference in signal similar to Fig. 7.8(C). Otherwise, the LFP is extremely similar per feature. This indicates that feature specific information is rare but can happen in the LFP.

7.3.5 Decoding the Search Period

This finally leads us to questions about the content of the saccade signal. One perspective is that there is some internal model that is accessed every time the subject saccades. Or, more generally, there is a computation after every saccade. If true, then there should be differences based on when the subject decides to continue or stop searching. The timeline would be the following: saccade to a new object, see the three features, perform a computation, then decide whether to search further or stop searching. One counter-possibility is that the subject decided to saccade to a specific object based on peripheral vision.

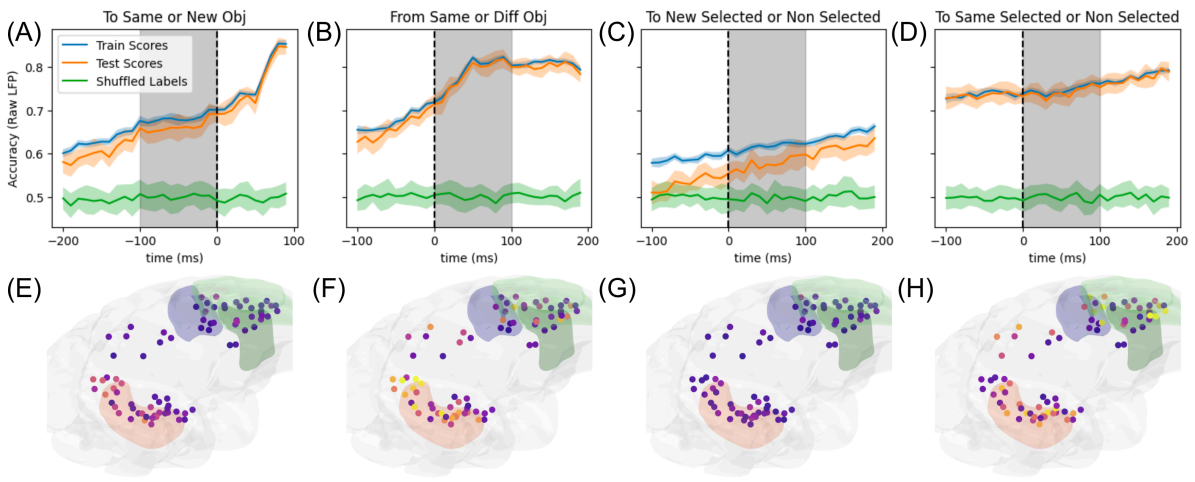

Figure 7.9: Train and test accuracy of decoders of differences in saccades. (A) Difference between the future saccade heading towards a new or the same object, beginning at $t = 0$ ms. (B) Difference in the past saccade heading towards a new or same object, ending at $t = 0$ ms. (C) Difference in when the previous saccade, to a new object, was to the selected or non-selected object. (D) Difference in whether the previous saccade, without leaving the object, remained on the selected or non-selected object. The associated coefficient weights are shown with the same colorscale in (E-H). The largest differences, in (F,H), show that the main weights during fixation (gray bands in (A-D)) occur close to the tip of the medial temporal lobe/hippocampus. Similar effects can be seen in various frequency bands (not shown) up to the Gamma range.

I don't believe this is the case, however, due to the lack of peripheral vision used (Fig. 3.17).

To answer the question of differences in accessing the internal model, I ask several questions. First, given a saccade to either the same object or to a new object, is there a difference in the period leading up to the saccade when deciding to stay on the object or leave the object? A follow up question can be asked, is there a difference if the previous saccade was to a new object or to the same object? The main difference is whether the time interval is before or after the saccade. Next, is there a difference between saccading to a new object, if the new object is the chosen or non-chosen object? Perhaps there is a difference when interpreting a selection or non-selection. Lastly, is there a difference when saccading to the same object, if that object is the selected or non-selected card? This may indicate a difference in signal when the subject knows to maintain fixation. Only non-stereotypical saccades (Sec. 2.9) are used. To stop decoding from future or past saccades, a 100 ms fixation is used and the important decoding accuracies should be within this region.

I use a linear support vector classifier ("dual=False" and $C = 10^{-7}$) on every 10th timepoint, locked to either the beginning or end of each saccade. Each "sample," associated with either 0 or 1 depending on the classification, is then a vector across all bipoles. The data is balanced by sub-selecting the same number of samples for classification. For 20 random 80/20 train-test splits, each sample is normalized based on the z-score calculated purely from the training split, unsuring no data-leakage. I also calculate chance by shuffling the sample labels. To examine the weights of the decoders across bipoles, I take the mean of the absolute value of coefficients across all splits and 100 ms fixation timings, ensuring only fixation times and ignoring the sign of the coefficients.

The first question of interest: given a 100 ms fixation on a non-selected card, is there a difference between saccading to a new object vs remaining on the current object? Perhaps the subject accesses their internal model, which dictates to keep observing the card or to switch to a new card. In Fig. 7.9(A), there is some decoding during the 100 ms fixation. Similar trends occur in lower frequency bands (not shown). Interestingly, if looking at the power between 65 and 87 Hz, there is decoding (65%) immediately before the saccade. The 65-87 Hz decoding may be due to action planning. The associated weights (Fig. 7.9(E)) show that this effect mainly occurs in the medial temporal lobe/hippocampus. Overall, this suggests that there are differences in the signal when planning on searching a new or same object.

A follow-up question is whether there is a difference based on the previous saccade. If the previous saccade was to a new object or remained on the same one, is there a difference for the 100 ms after saccading? In Fig. 7.9(B), there is a large amount of decoding accuracy, which happens in frequencies up to the Gamma range (not shown). The associated weights (Fig. 7.9(F)) show that this effect mainly occurs in the medial temporal lobe/hippocampus. Given this, there is likely visual processing of the new card after saccading.

Similar questions can be asked depending on if the saccade is to the selected or non-selected card. Given that the saccade was to a new object, is there a difference to saccading to the choice card or not? There may be a difference in processing when the subject settles on their choice. In Fig. 7.9(C), there is a very small amount of decoding, with small average coefficients (Fig. 7.9(G)), suggesting that there may not be large differences when the new card appears.

Lastly, there is the question of difference in visual content of the selected vs non-selected card. Given that the eye never left a card, is there a difference if the card was chosen or not? In Fig. 7.9(D), there is a substantial decoding accuracy, and the large coefficients (Fig. 7.9(H)) occur in the medial temporal lobe/hippocampus. Similar effects can be seen up into the Gamma frequency range. This suggests that there are differences during the choice and search period, which may be due to processing the choice.

Overall, there are differences mainly after the saccades occur. One can decode if the saccade switched objects or not, and one can decode if the card was the selected one or not. It's possible that the former case occurs due to processing of visual information, and its possible for the latter case to occur based on processing the choice. Further, it may take slightly longer than 100 ms to process a choice, aligning with the time it takes for visual signal to reach the PFC [140]. Computations are likely time-locked to the saccade, which could allow for further studies related to the visual information and behavioral models (Chapter 4).

Chapter 8

Non-Stationary Dynamical Mode Decomposition

A linear dynamical systems approach is useful in analyzing the LFP. In these approaches, one models the data as

$$\frac{d\mathbf{x}}{dt} = A\mathbf{x}. \tag{8.1}$$

With this, ideally a model is created that can probe characteristics of the data. For example, what information is present and where is it in the brain? How does this information flow? How do networks evolve in relation to behavioral events? What task variables are encoded? How do these representations change across a session or sessions? While many methods seek answers to these questions, none are completely satisfying. Latent Factor Analysis via Dynamical Systems (LFADS) [178], Generalized Linear Models, and Switching Linear Dynamical Systems [93, 179–181] are generally used on spiking data. Although, LFADS has recently been modified to apply to LFP [182]. Coherence [183], Granger Causality [184, 185], and Hilbert analysis [66, 186] work on LFP, but they are not model driven. There are also Dynamical Mode Decomposition methods [187, 188, 8, 189, 190], which provide a dynamical systems approach. However, they do not typically deal with non-stationary systems, a prevalent problem in LFP data.

Ideally, one would use a method that involves the frequency spectrum, is ideal for LFP, finds spatial and temporal modes, and temporally evolves. With this in mind, I developed a novel method: Non-Stationary Dynamic Mode Decomposition (NS-DMD), which generalizes standard stationary DMD approaches to non-stationary systems.

This work led to the published paper [1], presented here. The following peer-reviewed text, until Sec. 8.8, was co-authored by Ariel Rokem, Elizabeth Buffalo, Nathan Kutz, and Adrienne Fairhall. In Sec. 8.8, I present text on approximating modes from known systems, not included in the manuscript.

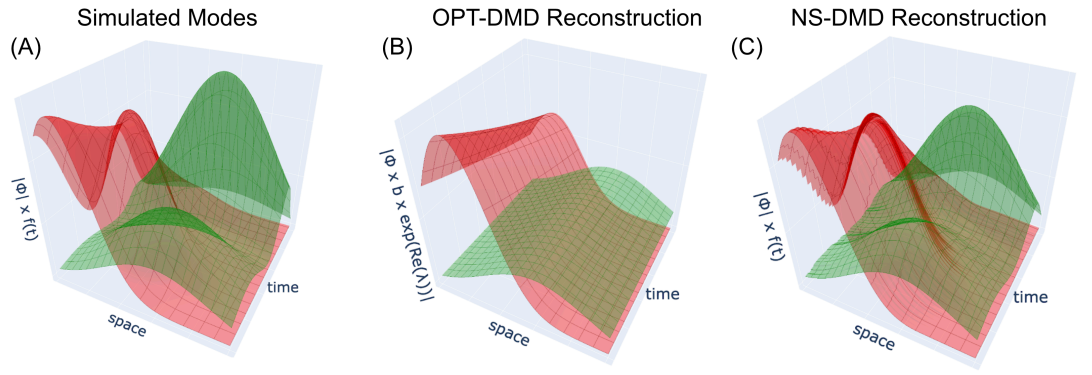

Figure 8.1: (A) The spatial distribution $|\phi_k|$ and global amplitude $f_k(t)$ of a simulated dataset, originating from the form $\mathbf{x}(t) = \sum_k \phi_k(t) \exp\left(i \int_{t_0}^{t} \omega_k(t) dt\right) f_k(t)$. The frequencies $\omega(t)$ are 20 Hz (red mode) and 30 Hz (green mode). The spatial modes mix together at the same time with varying amplitudes. The red modes spatial distribution fluctuates while the green mode turns on and off with irregular amplitudes. OPT-DMD fails to recover the original modes (B). NS-DMD samples the data with OPT-DMD during different time intervals, where dynamics are combined to find a subset of modes with modulating amplitude to reconstruct the data. NS-DMD recovers the original modes (C). Further exploration of NS-DMD and OPT-DMD on non-stationary datasets is shown in Fig. 8.3.

8.1 Abstract

Many physical processes display complex high-dimensional time-varying behavior, from global weather patterns to brain activity. An outstanding challenge is to express high dimensional data in terms of a dynamical model that reveals their spatiotemporal structure. Dynamic Mode Decomposition is a means to achieve this goal, allowing the identification of key spatiotemporal modes through the diagonalization of a finite dimensional approximation of the Koopman operator. However, these methods apply best to time-translationally invariant or stationary data, while in many typical cases, dynamics vary across time and conditions. To capture this temporal evolution, we developed a method, Non-Stationary Dynamic Mode Decomposition, that generalizes Dynamic Mode Decomposition by fitting global modulations of drifting spatiotemporal modes. This method accurately predicts the temporal evolution of modes in simulations and recovers previously known results from simpler methods. To demonstrate its properties, the method is applied to multi-channel recordings from an awake behaving non-human primate performing a cognitive task.

8.2 Introduction

Data-driven models of spatio-temporal systems are critical to understanding the evolution dynamics of natural systems and have become especially valuable given the increasing prevalence of large-scale measurements across all scientific disciplines. Many methods have been introduced to derive approximate dynamical models from data in domains such as fluid flows [191], climate systems [192, 193] and brain

activity [194–196]. However, in many data-driven approaches and algorithms, the data are assumed to be stationary when fitting the data. The stationarity assumption is violated in many datasets of interest, thus limiting potential model accuracy and forecasting capabilities. Deriving non-stationary generalizations of data-driven modeling is an area of active interest (e.g. [189, 190, 197]). We add to this effort by proposing a new method, *Non-Stationary Dynamic Mode Decomposition* (NS-DMD), which explicitly approximates the non-stationarity of the data while simultaneously constructing a low dimensional linear DMD approximation of multi-variate time-series.

NS-DMD builds on *Dynamic Mode Decomposition* (DMD) [8, 198], a systematic and unbiased method to reduce high-dimensional time-series data to a set of spatiotemporal modes. DMD approximates the Koopman operator [199], a linear infinite dimensional operator whose eigendecomposition models the observables that describe a finite dimensional, potentially non-linear, dynamical system [200–202]. In short, DMD approximates the data $\mathbf{x}(t)$ as

$$\mathbf{x}(t) \approx \sum_{k}^{r} b_k \phi_k e^{\omega_k t}, \tag{8.2}$$

where the ϕ_k are the DMD modes, ω are the DMD eigenvalues and b_k determines the weight of each mode. The modes are indexed by k up to a reduced rank r. The limitation of such an approach is the assumption of stationarity of the data. Simulated datasets of particular interest (Fig. 8.1(A)) are poorly reconstructed with stationary approaches (Fig. 8.1(B)) due to the spatial mixture of time-varying modes with time-varying amplitudes in the data. NS-DMD improves upon DMD by including time-dependence of the modes:

$$\mathbf{x}(t) \approx \sum_{k}^{r} b_k(t) \phi_k(t) e^{\omega_k(t) t}. \tag{8.3}$$

The additional time dependence in NS-DMD allows for accurate reconstruction of the underlying data (Fig. 8.1(C)). Further details on the NS-DMD method are found in Sec. 8.3.2.

Many approaches exist to fit non-stationary systems [93, 130, 179–181, 189, 190, 197, 203–209], but the approaches do not find representations in the form of Eq. 8.3. Related work is discussed in Sec. 8.5.1.

As with many other methods, NS-DMD assumes that the data are stationary in small time windows. We further assume that the data contain a low dimensional set of spectral components which may vary slowly with respect to their frequency. However, in contrast to previous methods, NS-DMD subsequently takes advantage of machine learning methods to associate modes across time windows, while systematically eliminating overfit and redundant modes. This allows us to detect global modulations of spatiotemporal modes that gradually drift across time.

We validate NS-DMD on synthetic data from several non-stationary systems. We then demonstrate its practical utility by analyzing multi-channel neuroscience time-series data. NS-DMD is able to

recapitulate results found using other more traditional time-series analysis methods, but also identifies non-stationary modes in these time-series data. We further demonstrate the utility by applying NS-DMD to sea surface temperature data, where NS-DMD recovers seasonal effects along with modes specific to the El Niño phenomena. Taken together, the novel findings and the connection to previous methods demonstrate the promise of NS-DMD for the analysis of non-stationary data.

8.3 Methods

Notation

We follow the notation in [210]. Scalars are denoted by lowercase letters (s), vectors by bold lowercase letters ($\mathbf{v}$), matrices by bold capital letters ($\mathbf{M}$), and tensors of third order by calligraphic bold letters ($\boldsymbol{\mathcal{T}}$). In summary:

- v_i denotes the ith entry of $\mathbf{v}$;
- m_{ij} denotes element (i, j) in $\mathbf{M}$;
- t_{ijk} denotes element (i, j, k) in $\boldsymbol{\mathcal{T}}$;
- $\mathbf{m}_{i:}$ and $\mathbf{m}_{:j}$ denote the ith row and jth column of $\mathbf{M}$;
- More compactly, $\mathbf{m}_j \equiv \mathbf{m}_{:j}$, denotes the jth column of $\mathbf{M}$;
- $\mathbf{t}_{ij:}$, $\mathbf{t}_{i:k}$, and $\mathbf{t}_{:jk}$ denote the vectors given by the corresponding free dimension of $\boldsymbol{\mathcal{T}}$;
- $\mathbf{T}_{i::}$, $\mathbf{T}_{:j:}$, and $\mathbf{T}_{::k}$ denote the matrices given by the corresponding free dimensions of $\boldsymbol{\mathcal{T}}$;
- More compactly, $\mathbf{T}_k \equiv \mathbf{T}_{::k}$ denotes the kth frontal slice;
- The nth element in a sequence is denoted by a superscript in parenthesis; e.g. $\mathbf{M}^{(n)}$ is the nth matrix $\mathbf{M}$.

8.3.1 Dynamic Mode Decomposition

Dynamic Mode Decomposition (DMD) [8] forms the backbone of NS-DMD. DMD approximates a low-dimensional representation of the data $\mathbf{X}$ in terms of linearly (exponential) evolving spatial modes (Eq. 8.2). That is, at fixed frequencies given by $\mathrm{Im}(\omega_k)$, there are spatial modes ϕ_k with loadings b_k that exponentially grow or decay. DMD is thought to combine the strengths of singular value decomposition across space with Fourier transforms across time.

There have been many improvements made to DMD since the original algorithm was introduced in 2008 [198], including a number of regression techniques for estimating the best fit linear dynamics [187, 188, 190, 211–213]. We build NS-DMD upon Optimized DMD (OPT-DMD) [187], which estimates the DMD modes and eigenvalues by using a variable projection optimization scheme

$$\text{argmin}_{\omega_k,\phi_k,b_k} \left\| \mathbf{X} - \sum_{k=1}^{r} b_k \phi_k \exp(\omega_k \mathbf{t}) \right\| \tag{8.4}$$

where the data is estimated up to a desired rank r. Optimized DMD iterates to a solution of this non-convex problem by using variable projection [214]. To improve convergence capabilities, often the exact DMD algorithm can be used as a seed for the initialization of the DMD algorithm. The OPT-DMD framework has been found to be the most robust algorithm to noise [187], providing an unbiased estimate of the DMD modes and eigenvalues for real data.

8.3.2 Non-Stationary Dynamic Mode Decomposition

For a data matrix $\mathbf{X} \in \mathbb{R}^{N \times M}$, DMD aims to accurately represent the data with a low dimensional set of K spatiotemporal modes $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N \times K \times M}$ as given by Eq. (8.4). When the governing processes vary in time, the set of spatiotemporal modes that best describe the data at one point in time may not describe the data well at another point in time. Furthermore, nonlinear dynamical systems may be better described locally by different linear approximations in different regions of phase space. We postulate that there are common dynamical modes that recur throughout an extended dataset. That is, we assume only a few modes exist that either vary in structure or vary in amplitude throughout time. The goal of NS-DMD is to find these recurring modes and weight their amplitude with time-dependent functions $\mathbf{F} \in \mathbb{R}^{K \times M}$ that characterize the time-varying contribution of each mode to a reconstruction of the data. Thus, NS-DMD seeks to approximate the data at any snapshot t_j with

$$\mathbf{x}_j \approx \sum_k \mathbf{s}_{:kj} f_{kj}. \tag{8.5}$$

To approximate $\boldsymbol{\mathcal{S}}$, a common approach [190, 197, 206, 215] is to split the data $\mathbf{X}$ of length M into W short overlapping windows of length $P < M$, $\tilde{\mathbf{X}}^{(w)} \in \mathbb{R}^{N \times P}$, defined by the corresponding set of time points $\mathbf{t}^{(w)} = \{t_1^{(w)}, t_2^{(w)}, ...t_P^{(w)}\}$. The processes governing the dynamical systems are assumed to be approximately stationary in each window, an assumption that is valid depending on the size of the window. If the size is too large, then the stationarity assumption is likely false. If the size is too small, there is a lack of statistical sampling to find reasonable solutions that may not generalize well across time. For example, in the limit of two snapshots, it is unlikely for modes to generalize to any another snapshots.

The data of each window are extracted with OPT-DMD [187], an iterative algorithm that finds r modes per window, where r is a chosen hyperparameter. For the first sampled window $\tilde{\mathbf{X}}^{(1)}$, OPT-DMD is executed without initial conditions or with educated guesses; OPT-DMD can automatically determine an initial guess if needed [187, 8]. For sampling windows $w > 1$, OPT-DMD is initialized with the normalized eigenvalues $\mathbf{\Lambda}^{(w-1)}/|\mathbf{\Lambda}^{(w-1)}|$ from the previous window, a process that favors spatiotemporal smoothness.

Having determined modes $\mathbf{\Phi}^{(w)} \in \mathbb{C}^{N\times r}$, $\mathbf{\Lambda}^{(w)} \in \mathbb{C}^{r\times r}$, and $\mathbf{b}^{(w)} \in \mathbb{R}^r$, local to every window w, the goal is to identify a set of modes that apply across the full M length dataset. A visual description of this process is shown in Fig. 8.2 (a). Formally, similar spatiotemporal modes are grouped into K groups across consecutive windows $\tilde{w} = \{w, w+1, ...w+n_k-1\}$, a process defined and explained in Sec. 8.3.2. We write this as $\tilde{\mathbf{\Theta}}^{(k)}$, where each $\tilde{\mathbf{\Theta}}^{(k)}$ contains a variable number n_k of modes $\{\theta^{(w)}, \theta^{(w+1)}, \theta^{(w+n_k-1)}\}$, and $\mathbf{\Theta}$ is a placeholder for $\mathbf{\Phi}$, $\mathbf{\Lambda}$, $\mathbf{b}$, and $\mathbf{t}$. To find time dependent quantities $\theta_j^{(k)} = h(\tilde{\mathbf{\Theta}}^{(k)})$ at time t_j, the function h averages $\tilde{\mathbf{\Theta}}^{(k)}$ during overlapping windows and extrapolates $\tilde{\mathbf{\Theta}}^{(k)}$ outside the range of $\mathbf{t}^{(k)}$:

$$\theta_j^{(k)} = \begin{cases} \frac{1}{N}\sum_{i\in\tilde{w}:t_j\in\mathbf{t}^{(i)}} \theta^{(i)} & t_1^{(w)} \le t_j \le t_P^{(w+n_k-1)} \\ \theta^{(w)} & t_j < t_1^{(w)} \\ \theta^{(w+n_k-1)} & t_j > t_P^{(w+n_k-1)}, \end{cases} \tag{8.6}$$

where N refers to the number of summed terms. We use a notation where $i \in \tilde{w} : t_j \in \mathbf{t}^{(i)}$ indicates all elements $i \in \tilde{w}$ such that $t_j \in \mathbf{t}^{(i)}$. A visualization of the function h is shown in Fig. 8.2 (b). The application of h leads to $|\phi_{:kj}| = h(|\tilde{\mathbf{\Phi}}^{(k)}|)$ and $\omega_{kkj} = h(\angle\tilde{\mathbf{\Lambda}}^{(k)})$. The angular part of $\mathbf{\Phi}^{(w)}$ defines the phase of every channel at the start of the respective window. To find $\angle\phi_{:kj}$, the phases are aligned to the start of the full dataset. For a mode k and window w_i, the phase is $\angle\phi_k'^{(w)} \equiv \angle\phi_k^{(w)} - \sum_{j=1}^{i-1}\omega_{kkj}\cdot$, and the phase at any time is $\angle\phi'_{:kj} = h(\angle\tilde{\phi}_k'^{(w)})$. To gain greater temporal smoothness, one can apply a moving average to the time dependent modes. The k time dependent spatiotemporal modes form the matrices $\mathbf{\Phi}_j \in \mathbb{C}^{N\times K}$ and $\angle\mathbf{\Lambda}_j \in \mathbb{C}^{K\times K}$ for every time point t_j. With the time dependent modes, the spatiotemporal modes $\boldsymbol{\mathcal{S}} \in \mathbb{R}^{N\times K\times M}$ are:

$$\mathbf{S_j} \equiv \mathrm{Re}(\mathbf{\Phi_j} \mathbf{e}^{\mathbf{i}\sum_{\mathbf{i=1}}^{\mathbf{j-1}} \angle\mathbf{\Lambda_i}}), \tag{8.7}$$

where the real part of each mode recovers the complex conjugate pairs in the definition of $\mathbf{S}_j$.

$\boldsymbol{\mathcal{S}}$ is weighted with an unknown temporal modulation $\mathbf{F} \in \mathbb{R}^{K\times M}$. A visualization of $\mathbf{F}$ is shown in Fig. 8.2(C). The time-series is reconstructed with

$$\hat{\mathbf{x}}_j = \sum_k \mathbf{s}_{:kj} f_{kj}, \tag{8.8}$$

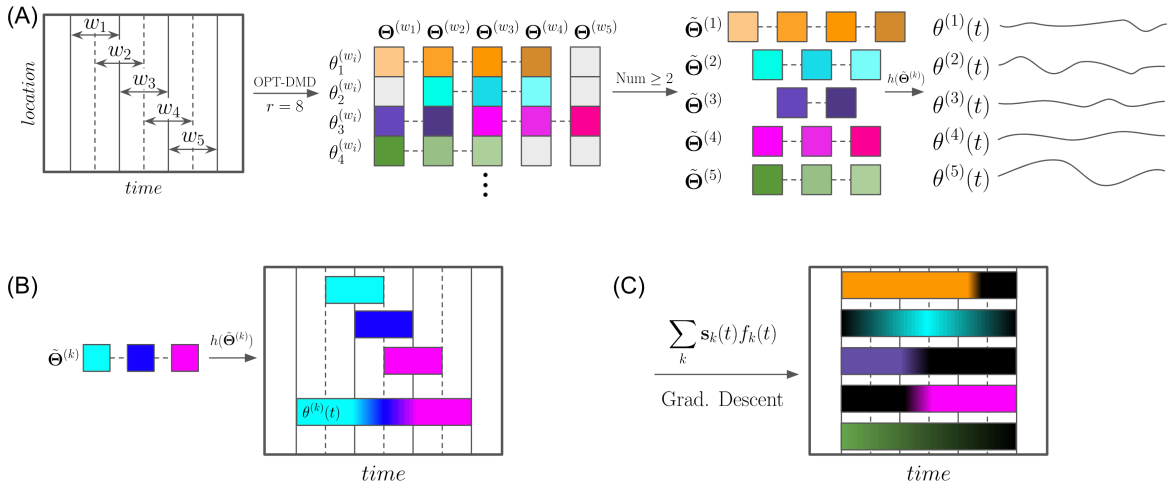

Figure 8.2: Flowchart of the NS-DMD algorithm. (A) A data matrix (first panel) is subdivided into windows. The number, size, and stride of the windows are hyperparameters. For each window, OPT-DMD is computed with r modes. The modes are visualized as squares in the second panel, where $\tilde{\mathbf{\Theta}}$ is a placeholder for the OPT-DMD modes: $\mathbf{\Phi}$ and $\mathbf{\Lambda}$. Modes are deemed similar to each other based on the procedure in Sec. 8.3.2; similar modes are connected by dashed lines and have similar colors. Different shades indicate that the modes may not be exactly the same. Groups of minimum 2 (another hyperparameter) similar modes (third panel) are transformed into continuous modes across time in the fourth panel via the function $h(\tilde{\mathbf{\Theta}})$. The function $h(\tilde{\mathbf{\Theta}})$ is visualized in (B). Similar, but slightly different modes, are indicated by slightly different colors. For example, the frequency could be $10Hz$ in window 2, $10.2Hz$ in window 3, and $10.4Hz$ in window 4. For overlapping windows, modes are averaged and extended outside the range of the windows. Following the example, the frequency would be $10.2Hz$ before window 3 begins, $10.1Hz$ during the overlap of windows 2 and 3, $10.3Hz$ during the overlap of windows 3 and 4, and $10.4Hz$ for the remainder. The modes $\theta^{(k)}(t)$ comprise the spatiotemporal modes $\boldsymbol{\mathcal{S}}$ (see text). Lastly, temporal modulations $f_k(t)$ of each mode $\mathbf{S}_k$ are found with gradient descent. $f_k(t)$ is visualized in (C), where the colored bars indicate when each mode well describes the data. $f_k(t)$ is flexible and can find gradual changes to the modes (e.g. the green and light blue mode). Other times, the modes turn on/off rapidly (e.g. the orange, purple, and pink modes). Lastly, the timing is flexible, indicated by the modes not necessarily turning on/off exactly at the dashed lines.

where the vectors $\hat{\mathbf{x}}_j$ form the estimated data matrix $\hat{\mathbf{X}} \in \mathbb{R}^{N \times M}$. To solve for $\mathbf{F}$, we use gradient descent (Sec. 8.3.2). An alternate, "exact" method is proposed in Sec. 8.7.1, which aims to directly solve Eq. 8.8. However, this method tends to be noisier than gradient descent.

Fitting Time-Varying Modes with Gradient Descent

Constraints are needed before applying gradient descent to find $\mathbf{F}$. First, $\mathbf{F}$ is non-negative; assemblies are either "on" with some variable amplitude, or they are "off." Second, a sparsity constraint is added since $\boldsymbol{\mathcal{S}}$ may contain redundant modes. Lastly, $\mathbf{F}$ is continuous since we assume that modes turn on or off on the timescale defined by the sampling rate $1/\text{sr} = \Delta t$.

The following loss function satisfies these constraints on $\mathbf{F}$:

$$L = \frac{1}{2}\left|\left|\hat{\mathbf{X}} - \mathbf{X}\right|\right|_F^2 + \alpha \sum_{k,j} |f_{kj}| + \frac{\beta}{2} \sum_{l=-N}^{N} \sum_{k,j} \left(f_{k,j} - f_{k,j+l}\right)^2 . \tag{8.9}$$

The first term is a standard least-squares loss term on the reconstruction of the data given $\mathbf{F}$. The α

term enforces sparsity in the solutions of $\mathbf{F}$. The absolute value will be dropped since $\mathbf{F}$ is forced to be non-negative after every iteration of gradient descent. Lastly, the β term enforces continuity across time; β controls the degree of smoothing, and the smoothing timescale is controlled by N. For simplicity, β is fixed for each l, although in principle β can fluctuate.

The gradient of $\mathbf{F}$ for each mode k and snapshot t_j is

$$\frac{dL}{df_{kj}} = (\hat{\mathbf{x}}_j - \mathbf{x}_j)\mathbf{s}_{:kj} + \alpha + \beta \sum_{l=-N}^{N} (f_{k,j} - f_{k,j+l}). \tag{8.10}$$

Finally, this allows us to compute the gradient descent for an iteration $i > 1$:

$$\mathbf{F}^{(i)} = \mathbf{F}^{(i-1)} - \gamma \frac{dL}{d\mathbf{F}}^{(i)} + \gamma\nu \frac{dL}{d\mathbf{F}}^{(i-1)}, \tag{8.11}$$

where γ is the learning rate and ν is the momentum [216].

The initial guess of f_{kj} is found by setting $\frac{dL}{df_{kj}} = 0$ and solving for $\alpha = 0$ and $\mathbf{f}_{k' \neq k, l \neq 0} = 0$:

$$f_{kj} = \frac{\mathbf{x}_j \mathbf{s}_{:kj}}{|\mathbf{s}_{:kj}|^2 + \beta(2N+1)}. \tag{8.12}$$

To remove noise, the initial guess is lowpass filtered.

After each iteration, all negative values of $\mathbf{F}$ are set to 0, and $2N+1$ consecutive values are averaged to further control the smoothness of $\mathbf{F}$. $\mathbf{F}$ is reflected at the boundaries. It is possible for edge artifacts to occur when $\mathbf{F} > 0$, so a minimum of N values should be trimmed at the boundaries.

After running gradient descent, the average amplitude of each mode is typically smaller than the true value due to α, β, and the averaging step. The amplitude, defined as $\mathbf{a} \in \mathbb{R}^K$, of each mode is adjusted with the least squares algorithm: $\mathbf{x}_j = \sum_k \mathbf{s}_{:kj} a_k f_{kj}$. The amplitude $\mathbf{a}$ is absorbed into $\mathbf{F}$.

Feature Selection

Typically, while estimating $\mathbf{S}$, one finds a large number of redundant modes, even with the sparsity constraint in gradient descent. We turn to feature selection to subselect modes.

To combine redundant spatiotemporal modes, the similarity of pairs $\mathbf{S}_i$ and $\mathbf{S}_{j\neq i}$ is determined. The frequencies, spatial amplitudes, and spatial phases are all needed to be similar for two modes to be defined as similar. The difference in frequencies $f = \angle\lambda/(2\pi)$ must be within a desired threshold: $|f_i - f_{j\neq i}| < \text{thresh}_1$. The cosine similarity $C(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A}\cdot\mathbf{B}}{|\mathbf{A}||\mathbf{B}|}$ between spatial amplitudes $|\phi|$ must be above a desired threshold: $C(|\phi_i|, |\phi_{j\neq i}|) > \text{thresh}_2$. The spatial phases $\angle\phi$ need to first be aligned since they are referenced to their window's initial t_0. To align, we define $\Delta\angle\phi \equiv (\angle\phi_i + 2\pi f_i \Delta t/2) - (\phi_{j\neq i} - 2\pi f_{j\neq i}\Delta t/2)$, where $\Delta t = t_{0,j\neq i} - t_{0,i}$. Due to periodicity, all $\Delta\angle\phi$ are shifted to within $-\pi$ to π. The spatial phase is similar if $\frac{1}{n}\sum_i^n \Delta\angle\phi_i < \text{thresh}_3$. The solutions $\mathbf{S}_i$ and $\mathbf{S}_{j\neq i}$ are considered similar (i.e., redundant) if

all three threshold checks are valid.

The first redundant set of modes are found from the parity of spatiotemporal modes. If the data is sufficiently approximated by sines and cosines, then OPT-DMD returns pairs of solutions $\mathbf{S}$ with opposite signs: $|\phi|e^{i\angle\phi}e^{i\angle\lambda t_i}$ and $|\phi|e^{-i\angle\phi}e^{-i\angle\lambda t_i}$. The real part of $\mathbf{S}$ is compared between pairs, where one mode is removed per pair.

Next, the modes from consecutive windows are compared. Since OPT-DMD is computed on each window using the eigenvalues of the previous window as an initial guess, the modes typically remain similar across time unless the assembly drastically changes. In some cases, the frequency or spatial distribution of modes may fluctuate over time. A method that is very sensitive to such changes would generate additional new modes. Our modeling goal is to have a single mode describe the fluctuation, so the method needs to be flexible when the frequency or spatial distribution drifts in time. This is later tested with great success in Sec. 8.4.1. If modes are similar across time, we stitch them together according to Eq. 8.7, as described in Sec. 8.3.2.

Next, if desired, one can retain only groups of modes that have more than a user-defined minimum number of consecutive similar windows. This enforces that $\boldsymbol{\mathcal{S}}$ is somewhat consistent across time. Then, the reconstruction error is calculated for each mode independently, defined as the cosine distance between $\mathbf{X}$ and the reconstruction $\hat{\mathbf{X}}$. The cosine distance is chosen since it does not depend on the amplitude of each mode. A user-defined number of modes or any modes with a reconstruction error above a user-defined threshold are retained. This drastically reduces the number of modes to ones that generally reconstruct the data well.

After initially reducing the number of modes, standard feature selection methods [217] are used to find the subset of the remaining modes that best reconstruct the data. The basic feature selection algorithms of [218] have been implemented. These methods start with either none or the entire set of modes and add/subtract one mode at a time while checking the reconstruction error. We first run gradient descent (Sec. 8.3.2) while adding/removing each mode independently. Then, the mode that decreased the cosine distance the least is removed. The process is repeated until a final, user selected number of modes have been added/removed. The best sub-selection of modes can be chosen from an "elbow" curve of overall cosine distance as a function of number of modes.

Sampling from a Broad Frequency Range

A dataset may contain a large number of modes at many different frequencies. Running OPT-DMD with many modes can be slow and inaccurate if some frequency bands have smaller amplitudes. To compensate, an additional step to NS-DMD is to bandpass over many different frequency ranges. Initial guesses of frequencies should lie within the bandpass ranges, and OPT-DMD is computed for each window in each band. Only solutions where the frequency is within the bandpassed range are included. All modes

from all bandpassed ranges are combined in the feature selection step. Gradient descent is ran on the original, non-bandpassed data.

Either a type I Chebyshev filter of order 5 and a pass-band gain of 1 or a Butterworth filter of order 5 is used to bandpass filter the data. Due to edge artifacts at the temporal extremes of the data, it is recommended to bandpass for a longer time range than of interest and trim the excess timepoints.

To determine the accuracy of the reconstruction during sequential feature selection methods, it's best to calculate the reconstruction similarity for each frequency band:

$$\frac{1}{N}\sum_{i=1}^{N} C(\mathrm{BP}_i(\mathbf{X}), \mathrm{BP}_i(\hat{\mathbf{X}})), \tag{8.13}$$

where $\mathrm{BP}_i(\mathbf{A})$ is a function that bandpasses the data $\mathbf{A}$ to the ith frequency range. This method of calculating the reconstruction similarity does not preferentially bias toward modes that occur with a comparatively large amplitude.

Non-Stationary Dynamic Mode Decomposition Algorithm

Algorithm 1 NS-DMD

Input: $(\mathbf{X}, \mathbf{t}, \text{sampling rate}, \text{hyperparameters})$
Procedure: NS-DMD$(\mathbf{X}, \mathbf{t}, \text{sampling rate})$

Bandpass($\mathbf{X}$) into bands (Optional)
for band and window **in** bands and windows **do**
    Compute $\mathbf{\Phi}$, $\mathbf{\Lambda}$, $\mathbf{b}$ with OPT-DMD
end for
for mode i and window j **in** modes and windows **do**
    Calculate $f_i^{(j)}$, $|\phi_i^{(j)}|$, and $\Delta\angle\phi_i^{(j)}$
    if $j = 0$ **or** (
        $|f_i^{(j)} - f_i^{(j-1)}| < \text{thresh}_1$ **and**
        $C(|\phi_i^{(j)}| - |\phi_i^{(j-1)}|) > \text{thresh}_2$ **and**
        $\frac{1}{n}\sum_k^n \Delta\angle\phi_{ik}^{(j)} < \text{thresh}_3$
) **then**
        make new group with mode j
    else
        mode j joins group with mode $j-1$
    end if
end for
Reject groups where size < minimum size (optional)
Stitch groups of modes to define $\boldsymbol{\mathcal{S}}$
Reject modes with worst reconstruction error (optional)
Run forward/backward algorithm to reject modes (optional)
Run gradient descent to get $\mathbf{F}$
Trim edge artifacts
Find overall amplitude with least squares

We conclude this section by summarizing the full method. An algorithmic version is given in Algorithm 1.

- Step 1 (optional): If the data has a large number of modes or includes modes with a much smaller amplitude, we bandpass the data into many different small bands. To evaluate the necessity, we suggest analyzing the power spectral density.
- Step 2: Run OPT-DMD for every window of interest and for every frequency band if applicable.
- Step 3: Find the similarity of consecutive modes. Group modes if they meet thresh_1, thresh_2, and thresh_3 requirements.
- Step 4 (optional): Keep only solutions that are similar for a user defined number of windows.
- Step 5: define $\boldsymbol{\mathcal{S}}$ based on the similarity of consecutive modes. Optionally include temporal lags for each recording location.
- Step 6 (optional): Initially reduce the number of modes by finding the reconstruction error of each mode independently.
- Step 7 (optional): Run the forward/backward elimination algorithm.
- Step 8: Run gradient descent on the final subset of modes. Make sure that this is done on the non-bandpassed data.
- Step 9: trim the data to remove edge artifacts.
- Step 10: use a least squares algorithm to find the final estimate of $\mathbf{F}$.

8.4 Results

We tested NS-DMD on simulations, electrophysiological brain data from multichannel recordings of local field potentials in the macaque brain, and from sea surface temperature (SST) data. NS-DMD recovers the underlying dynamics in the simulations. In the electrophysiological brain data, we show a rich set of modes that are active during different periods of performance of a cognitive task. In the SST data, NS-DMD recovers seasonal modes along with El Niño modes. Hyperparameters are described for all applications in Appendix Sec. 8.6.2. Further simulations, which require optional steps of NS-DMD or are of interest to specific communities, are included in the supplementary material (Sec. 8.7.3).

8.4.1 Simulations

Synthetic data is generated from the following generic equation:

$$\mathbf{x}(t_i) = \sum_k f_k(t_i)\phi_{A,k}(t_i) * \cos(\sum_{j=0}^{i-1} \omega_k(t_j)\Delta t + \phi_{P,k}(t_i)), \tag{8.14}$$

where $\mathbf{x}(t)$ is the data consisting of a vector of channels, $f_k(t)$ is the amplitude modulation of all channels, $\omega(t)$ is the time varying angular frequency, $\phi_{A,k}$ is the time varying normalized amplitude ($|\phi| \equiv 1$), $\phi_{P,k}$ is the time varying phase, and $\Delta t \equiv 0.001$ is the time delay between snapshots. This is the more explicit form of the simplified form of the data in Eq. 8.5, which can be seen by expressing the cosine in terms of exponentials and combining with $\phi_{A,k}$ to form $\mathbf{S}$.

Non-Stationarity in Multiple Assemblies

The simplest case of interest is when multiple assemblies switch on/off with non-constant amplitudes, corresponding to the following simplification of Eq. 8.14:

$$\mathbf{x}(t_i) = \sum_k f_k(t_i)\phi_{A,k}\cos(\omega_k t_i + \phi_{P,k}). \tag{8.15}$$

We construct a group of modes that are active in the beginning, a group of modes that are active towards the end, a switching period where multiple modes coexist, and a mode with non-constant amplitude. The assemblies, labeled from A to D, construct the data shown in Fig. 8.3A. Assemblies A and B operate with a constant amplitude for 1700 ms, assembly C turns on with a constant amplitude at t =1300 ms, and assembly D's amplitude follows the shape of a Gaussian distribution starting at t =1300 ms. All assemblies initiate and decay with a fixed timescale. The exact shape of $\mathbf{F}$ is shown in Fig. 8.3D.

The frequency $\omega_k/(2\pi)$ is 4 Hz, 30 Hz, 17 Hz, and 30 Hz for modes A-D respectively. Assemblies B and D have the same frequency, but with different spatial distributions. For A and B, the spatial amplitude is $\phi^{1-50}_{A,\{k=A,B\}} = 1/N$ and $\phi^{51-100}_{A,\{k=A,B\}} = 2/N$, where the superscript labels the channel number and N is a normalization such that $|\phi_{A,k}| = 1$. The amplitudes of assemblies C and D are reversed such that $\phi^{1-50}_{A,\{k=C,D\}} = 2/N$ and $\phi^{51-100}_{A,\{k=C,D\}} = 1/N$. The exact amplitudes are shown in Fig. 8.3E. The spatial phases $\phi_{P,k}$ for every assembly form ten groups of ten channels. The phases are constant within each group of channels. Assembly A has a temporal delay $\phi_{P,k}/\omega_k$ from 0 ms to 30 ms across all groups; Assembly B has a delay from -20 ms to 30 ms; Assembly C from 50 ms to 0 ms; and Assembly D from 30 ms to -10 ms. The phases are shown in Fig. 8.3F. Each channel receives independent white noise with a standard deviation of 0.1.

For the majority of windows, there are two assemblies that follow a cosine function or two exponential functions. Thus, the minimal complete number of OPT-DMD modes per window is 4. We run NS-DMD with 4 modes per window, and the reconstruction of the data is shown in Fig. 8.3B. The reconstruction error is $\sqrt{\text{MSE}} \approx 0.1$ (Fig. 8.3J), the same as the added noise. Comparing the true and fitted $\mathbf{f}$ in Fig. 8.3D, NS-DMD recovers the correct mode amplitudes, including the non-stationary amplitude in assembly D. The spatial amplitudes and phases in Figs. 8.3E and 8.3F show recovery of the correct underlying modes. The inaccuracy in the 30 Hz (green) mode in (F) occurs because the phase is compared at $t = 0$

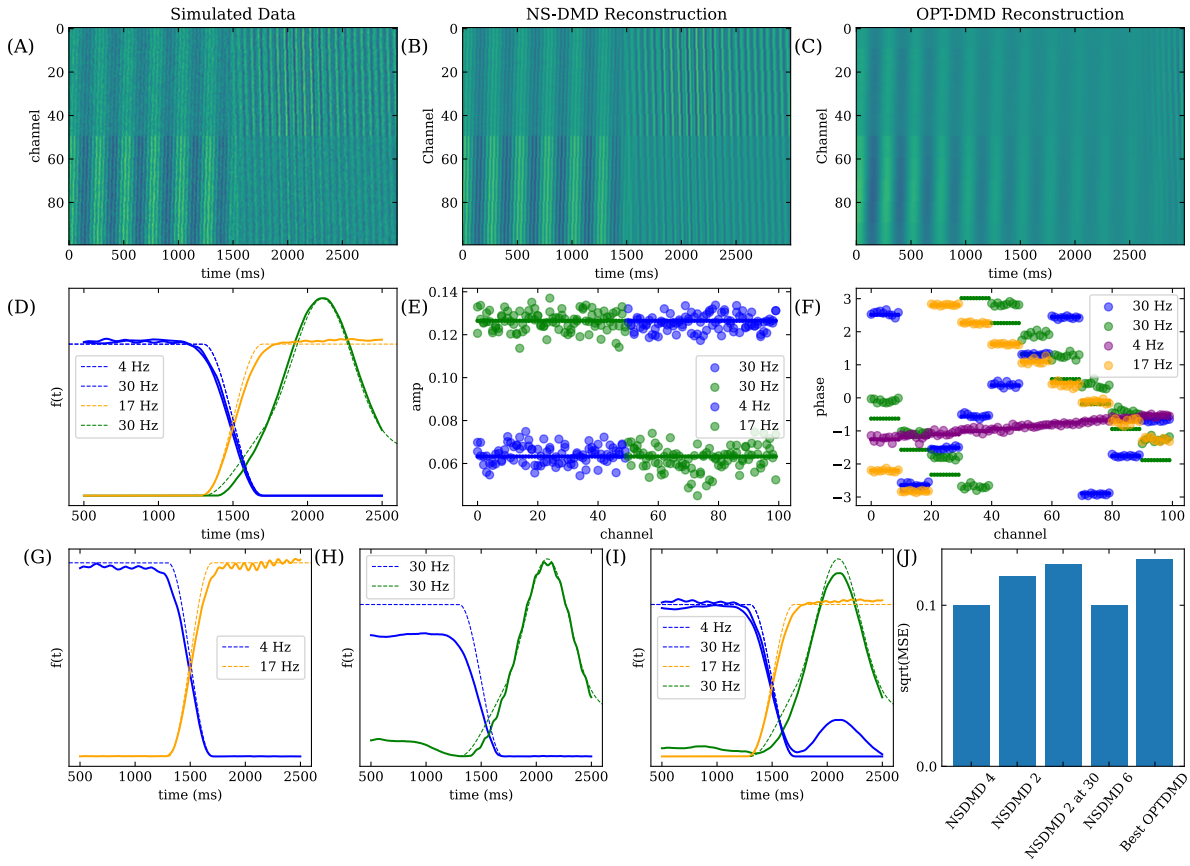

Figure 8.3: (A) The simulated data, generated from 4 assemblies with frequencies of 4 Hz, 30 Hz, 17 Hz, and 30 Hz. The dotted lines in (D) show the modulation $f(t)$ for each assembly, and the small solid circles show the amplitude and phase of $\mathbf{\Phi}$ in (E) and (F). Gaussian noise with a standard deviation of 0.1 is added independently to each channel and snapshot. (B) shows the reconstruction of the data with NS-DMD with 4 modes per window. (C) shows the best reconstruction with OPT-DMD (rank of 4). (D) compares the true (dotted lines) and fit $\mathbf{F}$ (solid lines) for each mode/assembly. (E) and (F) compares the true (small, solid dots) and fitted (large, transparent dots) $\mathbf{\Phi}$ amplitudes and phases for each channel. From this, we see the accuracy of the underlying modes, $f(t)$, and reconstruction of the data when using NS-DMD with an optimal number of 4 modes per window. We compare the the true (dotted lines) and fit $\mathbf{F}$ (solid lines) when running NS-DMD with 2 (G), 2 and an initial guess of ± 30 Hz (H), and 6 (I) modes per window. We find that 2 modes per window fits lower frequency modes well. Forcing the initial guess to higher frequency modes, as in (H), leads to higher deviations from the ground truth. Using 6 modes per window, as in (I), leads to overfitting at 2000 ms, which may be expected due to the similarity of the constructed 30 Hz modes. The reconstructed error for each number of modes per window is shown in (J), where the error is on the order of the Gaussian noise. NS-DMD obtains the best performance with either 4 or 6 modes per window.

ms; the phase is most accurate at $t \approx 2000$ ms when the mode is active, and any small inaccuracy in frequency will grow when analyzing the phase at a large time difference away. Overall, this simulation shows that NS-DMD can capture the underlying modes and temporal variations of a non-stationary linear dynamical system.

In practice one does not know the optimal number of modes per window. If instead we run NS-DMD with a smaller number of modes than optimal, 2 per window, it will only be able to capture half of the modes at any timestep. Due to the switching of assemblies, we expect NS-DMD to recover one early assembly (A or B) and one late assembly (C or D). NS-DMD recovers the 4 Hz and 17 Hz mode, as shown in Fig. 8.3G. This occurs due to the procedure; in the first window after the 4 Hz mode "turns off," the initial guess with OPT-DMD is still 4 Hz. OPT-DMD then converges to the closest solution, which happens to be 17 Hz. The reconstruction error (Fig. 8.3J), is marginally worse, as expected since only half of the modes are captured.

If the initial guess for the first window is forced to be 30 Hz, then NS-DMD finds the two 30 Hz modes (Fig. 8.3H)). NS-DMD performs noticeably worse when finding $\mathbf{F}$, but it still finds the correct trend. The low amplitude bias of the 30 Hz mode occurs due to a lack of an estimated 4 Hz mode, which can bias $\mathbf{X}$ to more positive or negative values for short (< 50 ms) windows. We verify this by rerunning this model with only assemblies B and D, where $\mathbf{F}$ is found correctly. The reconstruction error in Fig. 8.3J indicates a preference for lower frequency modes.

If we run NS-DMD with 6 modes per window, NS-DMD recovers 4 modes that reconstruct the data. The fitted and recovered amplitudes $\mathbf{F}$ in Fig. 8.3I show that NS-DMD performs well. Around t=2000 ms, both 30 Hz modes have a non-zero $f(t)$ which occurs due to their similar construction. The reconstruction error is on par with the amount of noise (Fig. 8.3J).

Lastly, we run OPT-DMD with ranks ranging from 2 to 10 and find the best performance with a rank of 4. The reconstruction is shown in Fig. 8.3C, where the original structure in the simulated data is blurred. Despite this, the reconstruction error is fairly small (Fig. 8.3J). However, the reconstruction error is worse than all NS-DMD models.

Smoothly Varying Time Dependent Modes

We now consider the case of smoothly varying time dependent modes (Fig. 8.4A). Allowing a time-varying frequency corresponds to the following simplification of Eq. 8.14:

$$\mathbf{x}(t_i) = \sum_k f_k(t_i)\phi_{A,k}\cos(\sum_{j=0}^{i-1}\omega_k(t_j)\Delta t + \phi_{P,k}). \tag{8.16}$$

We simulate three assemblies whose frequencies vary linearly in time around 17 Hz, 27 Hz and 33 Hz, where the true frequency $\omega_k/(2\pi)$ is shown in Fig. 8.4C. The 27 Hz assembly occurs early in time while

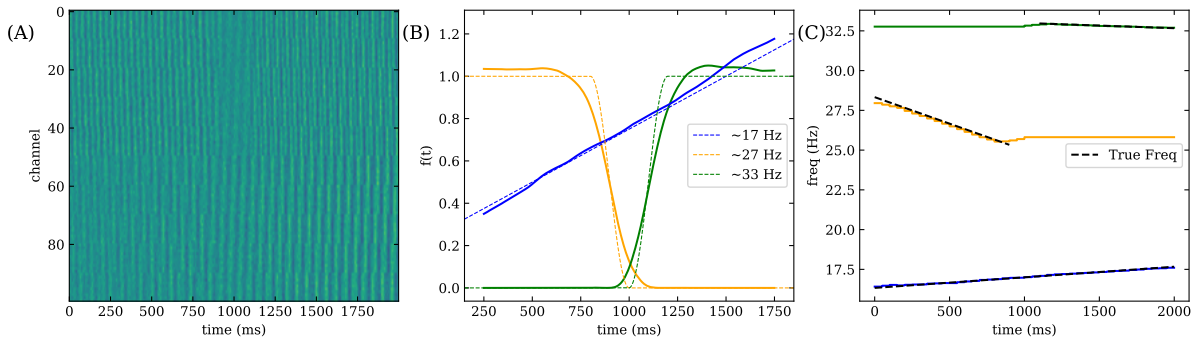

Figure 8.4: (A) Synthetic data for the smoothly varying time dependent mode simulation. (B) A comparison of the true (dotted lines) and fitted $\mathbf{F}$ (solid lines) for three assemblies with time dependent frequencies. (C) Comparison of the true frequencies (dotted lines) with the fitted frequencies (solid lines) for each drifting assembly. Note that the frequency is undefined when $\mathbf{F} = 0$. NS-DMD recovers the true modulations and frequencies.

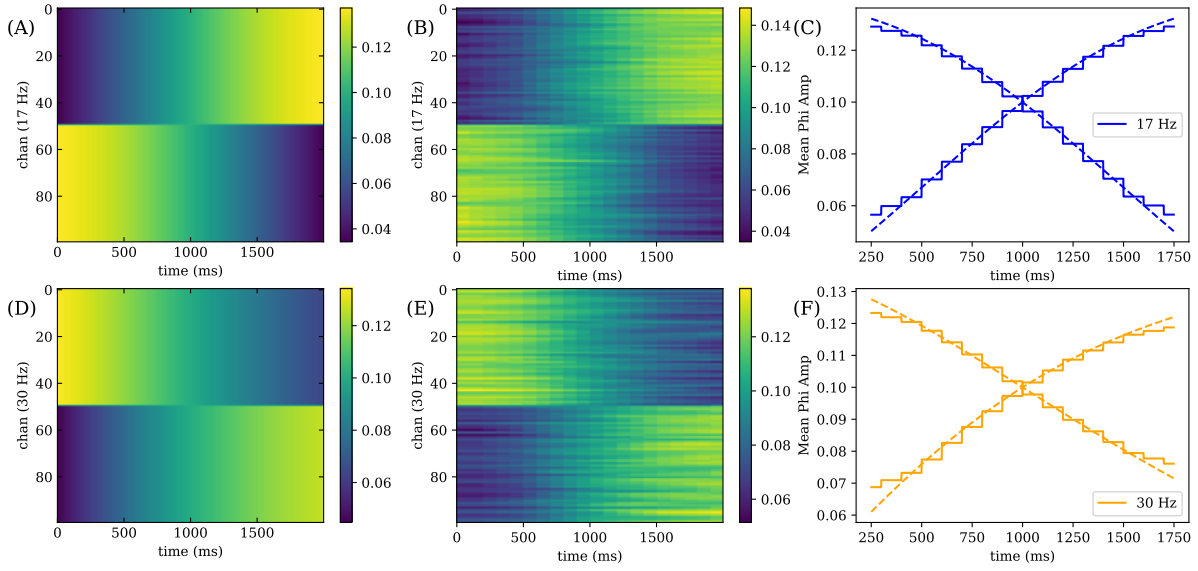

Figure 8.5: (A,D) True spatial amplitude distribution for time dependent spatial modes at 17 Hz and 30 Hz, where channels 1-50 and 51-100 are grouped and vary the same way. (B,E) Fitted spatial amplitude distribution for the 17 Hz and 30 Hz assemblies. (C,F) Comparison of the true (dotted) and fitted (solid) mean ϕ amplitude for channels 1-50 and 51-100. We find that NS-DMD recovers the correct spatial amplitudes $\phi_{A,k}$, where the staircase pattern occurs due to the 100 ms stride of NS-DMD.

the 33 Hz assembly occurs late. The 27 Hz assembly appears with a linearly varying amplitude for the duration of the dataset. The exact amplitudes $\mathbf{f}(t)$ are shown in Fig. 8.4B.

NS-DMD recovers the correct $\mathbf{F}$ (Fig. 8.4A), and time dependent frequency (Fig. 8.4B). Note that the frequency is undefined when $f(t) = 0$, indicated in Fig. 8.4B by ending the dashed lines.

We next allow for the spatial amplitude to fluctuate instead, corresponding to the following simplification of Eq. 8.14:

$$\mathbf{x}(t_i) = \sum_k f_k(t_i)\phi_{A,k}(t_i)\cos(\omega_k t_i + \phi_{P,k}). \tag{8.17}$$

The frequencies of two assemblies are fixed to 17 Hz and 30 Hz. The spatial amplitude is shown in Figs. 8.5A and 8.5D respectively, where channels 1-50 and 51-100 are grouped together with the same change in $\phi_{A,k}$ over time.

NS-DMD recovers the correct spatial amplitudes, shown in Figs. 8.5B and 8.5E with a step like variation due to discrete windows. To aid visualization, we plot the average amplitude for each mode and group of channels, 1-50 and 51-100 in Figs. 8.5C and 8.5F respectively.

We run OPT-DMD as well for both simulations of fluctuating frequency and spatial modes. The reconstruction error for both cases is worse for OPT-DMD at about 0.12 compared to 0.1 for NS-DMD. Like in the example in Fig. 8.3, the visual features are blurred with OPT-DMD (not shown), indicating the need for NS-DMD to recover the correct underlying modes.

8.4.2 Application to Electrophysical Brain Data

We apply NS-DMD to local field potentials (LFP) [100], an invasive measurement technique in which electrodes measure electric potentials deep inside the brain. In general, the LFP power decays approximately as a power law, with an exponent between -1 and -2 [219]. A wealth of literature has found correlations between behavioral parameters and time-varying power in various frequency bands of the LFP [104, 9, 220]. Other research suggests the possibility of cross-frequency coupling [39, 221–223] as a top-down mechanism of control.

The LFP is typically analyzed using standard time-series procedures, such as Hilbert or spectrogram analysis [66, 186], coherence analysis [183], and Granger Causality [184, 185]. While these methods are useful for understanding the structure of the data, they do not lead to a dynamical systems model of the brain. Others have argued for the use of a Koopman operator approach [224]. A DMD approach has been applied to sleep activity [215], revealing sleep spindle networks. Given its non-stationary spectral properties and the potential for the application of DMD in analysis of brain activity, LFP data are a perfect candidate for NS-DMD.

We demonstrate that NS-DMD can find consistent, repeatable spatial modes that co-activate intermittently in correspondence with the task. Modes activate and deactivate in correspondence with task

events, and they cluster in different areas of the brain. Further, some clusters show consistent phase differences between brain areas, indicating information flow. NS-DMD is also able to recover results from standard time-series analysis. We apply our methods to LFP data collected in the Buffalo lab from a macaque monkey performing the variant of the Wisconsin Card Sorting Test [95].

Dataset

We apply our methods to LFP data collected in the Buffalo lab from a macaque monkey performing a variant of the Wisconsin Card Sorting Test [95]. Out of 4 non-human primates performing the task, two have electrodes (FHC and Alpha Omega) implanted for neural recordings. A single subject is chosen for analysis and comparisons between methods. The subject is an adult female rhesus (Macaca Mulatta), aged 9 with a weight of 9.1 kg. The subject was headfixed with a titanium rod in a dimly lit room. The subject was positioned 60 cm away from a 19-inch CRT monitor, with 33 degrees by 25 degrees of visual angle. Stimuli were presented on the screen with software (NIMH Cortex). All procedures were carried out in accordance with the National Institutes of Health guidelines and were approved by the University of Washington Institutional Animal Care and Use Committee.

Trials are initialized when the animal fixates on a cross in the middle of the screen. The monkey must then choose one correct image out of four simultaneously presented images based on an uncued rule. The rule is discovered by trial and error. Each image has one of four possible shapes, colors, and patterns, and the rewarded rule is one of the 12 possible visual features. Animals are rewarded with a juice/chow mixture for 1400 ms if correct, and given a 5000 ms timeout period if incorrect. The rule remains fixed across consecutive trials while the monkey learns it; after 8/8 or 16/20 correct trials, the rule spontaneously changes.

The LFP is recorded for ~ 3 hours per session across several months using 220 electrodes implanted in multiple locations throughout the brain, including hippocampus and prefrontal cortex; we focus here on data from a single day. We ignore 17 electrodes that are dominated by noise, determined based on an unusually large amount of 60 Hz wall noise. We neglect trials where the remaining electrodes experience random artifactual bursts, or LFP activity that reaches the maximum or minimum possible recording limit of the electrode. This leaves us with 896 trials recorded on 203 electrodes to analyze. NS-DMD is applied to the raw data, normalized by z-scoring each electrode's signal in the 1-40 Hz range independently for each trial.

NS-DMD on a Single Trial

We fit NS-DMD on a single trial of the LFP and analyze the resulting modes. There are non-zero modes during every section of the task. The amplitude $f(t)$ of up to 10 modes within the 2-7, 12-17, 22-27, and 32-37 Hz modes are plotted in Fig. 8.6(A). Other modes exist, but are not shown for concision. Some

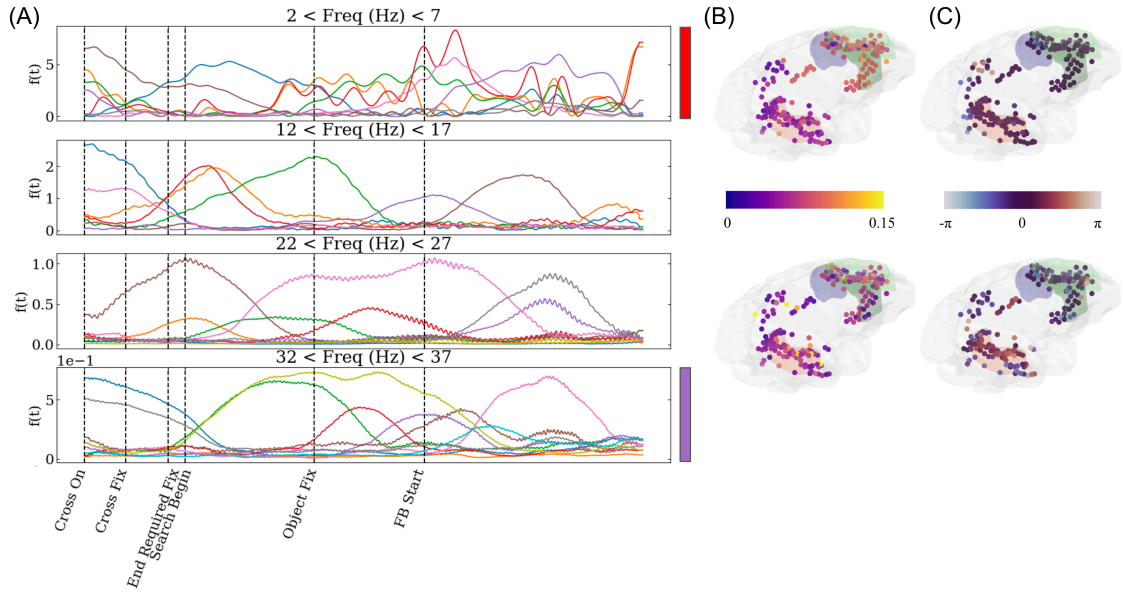

Figure 8.6: Example NS-DMD modes for the LFP data. (A) shows the mode amplitudes $f(t)$, separated by frequency ranges: 2-7, 12-17, 22-27, and 32-37 Hz. Modes persist for varying lengths of time. (B) shows the absolute value of the relevant spatial amplitudes $|\phi|$ for the red (top) and purple (bottom) modes. There are large spatial spreads for each mode. (C) shows the relevant spatial phases $\angle\phi$ for the red (top) and purple (bottom) modes on a circular color scale. There are clear divides between the phases of different brain areas, indicating that some areas lead or lag behind others. Shaded regions are the hippocampus (red) and prefrontal cortex (green), which are important for decision making and memory.

modes span very long stretches of the trial, while others turn on or off relative to task events. The red mode of the top plot and the purple mode of the bottom plot of Fig. 8.6(A) are selected for further analysis. In Fig. 8.6(B) and Fig. 8.6(C) the spatial amplitudes $|\phi|$ and phases $\angle\phi$ are shown for the red mode (top) and purple mode (bottom). The spatial amplitude $|\phi|$ is plotted. The spatial modes showcase widespread patterns across the brain. In Fig. 8.6(C), the phases show significant differences across various areas of the brain, indicating that some areas lead or lag behind other areas in these particular modes. In particular, there is a consistent phased difference between the red and green shaded regions in the purple mode, indicating information flow between the two areas.

Common NS-DMD Modes in All Trials

We then apply NS-DMD on all trials for 1500 ms after feedback begins, focusing on the differences between correct and incorrect trials in the 19-21 Hz frequency band. We perform k-means clustering [225] on the mean spatial amplitude $|\phi|$, where each "point" in the k-means algorithm is the average $|\phi|$ when $f(t) > \text{mean}(f(t))$. We choose 3 clusters for both "correct" and "incorrect" modes. We further separate the data by performing k-means clustering with 3 modes using the temporal amplitudes $\mathbf{F}$ for each previously found group. This separates the modes into clusters that have distinct spatial distributions and temporal distributions. We select three groups out of the nine for both correct and

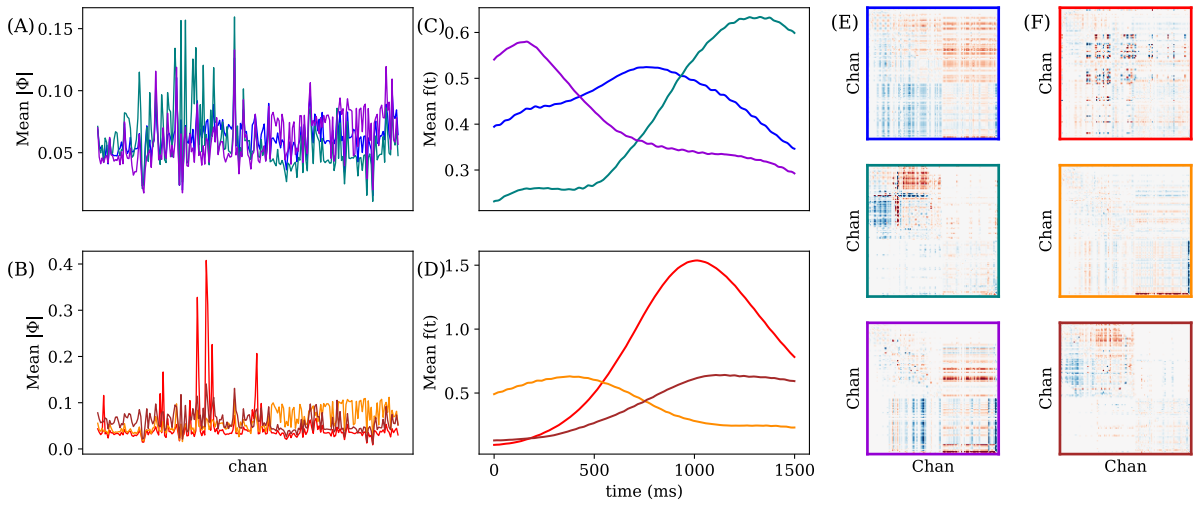

Figure 8.7: We group all modes corresponding to correct or incorrect trials into 3 clusters with K-means on the spatial amplitude. We further separate each cluster into 3 additional clusters based on $f(t)$ for a total of 9 groups per correct and incorrect trials. 3 groups (consistent colors) are selected for plotting, emphasizing consistent information flow during different periods after feedback. (A) and (B) show the average spatial amplitude of each cluster for the correct and incorrect clusters. (C) and (D) show the mean global modulations $\overline{f(t)}$ for each correct/incorrect cluster. Note that $\overline{f(t)}$ is averaged over many modes with independent $f(t)$'s. Thus, local maximums indicate when some, but not necessarily all, modes are large. (E) and (F) show the average phase difference between every pair of electrodes for correct/incorrect clusters. The phase difference is set to 0 for every channel pair where the average phase amplitude is less than 0.4, corresponding to a p-value less than 3×10^{-6}. The color limits are from $-\pi/4$ to $\pi/4$. Large amounts of widespread information flow occur immediately after feedback and late after feedback during correct trials.

incorrect trials for plotting.

The average spatial amplitude for each cluster is shown in Figs. 8.7A and 8.7B. Different clusters are associated with activity in different channels; some clusters have large amplitudes in single channels. The average global modulation $\overline{f(t)}$ of each "correct"/"incorrect" cluster is shown in Figs. 8.7C (correct) and 8.7D (incorrect). In the correct trials, **F** separates into early and late modes. In the incorrect trials, there is a cluster with a large amplitude 1s after feedback is given. The overall amplitude of the "incorrect" modes is much larger than the "correct" modes.

We are interested in the phase difference between each channel pair, since this is indicative of information flow. The phase difference is averaged separately across correct and incorrect trials when $f(t) > \overline{f(t)}$. We represent the phase difference θ as a complex vector on the unit circle and average:

$$\overline{\theta} = \sum_i e^{i\theta_i}. \tag{8.18}$$

The amplitude of the average phase difference vector indicates how consistently the two channels are related. We focus on pairs of electrodes with large amplitudes where $|\overline{\theta}| > 0.4$. Significance is calculated from the work of [137]. For 76 vectors, which is the minimum number of vectors averaged in Fig. 8.7, a threshold of 0.4 corresponds to a p-value of 3×10^{-6}. The average thresholded phase differences are shown in Figs. 8.7E and 8.7F. The "correct" clusters have the largest phase differences, suggestive of information

flow between two groups of channels after correct feedback is given. One incorrect cluster (red) has sparse consistent phase differences, which occur during incorrect trials one second after feedback. There are two clusters, green and dark-red, which have very similar average patterns: they are similar across space and time, and the phase differences are shared. This shows that a common pattern emerges for both correct and incorrect trials.

Comparison to Standard Analyses

To demonstrate the ability for NS-DMD to recover results from simpler methods, we compute analysis based on the Power Spectral Density, Hilbert analysis, and coherence.

First, for each electrode, we calculate the Power Spectral Density (PSD) to find the standard power law decay [100] using the Welch algorithm [226]. This results in the standard frequency power law in Fig. 8.8A. For NS-DMD, we average $f(t)$ across all trials and times within overlapping 3 Hz frequency bands (Fig. 8.8B). The power law is recovered with a similar slope.

Second, we compare NS-DMD to a traditional Hilbert analysis, where one typically analyzes a frequency band of interest. We bandpass the data to 27-37 Hz, Hilbert transform every channel, and take the absolute value of the resulting signal.We concatenate the trial and time dimensions, and run Principal Component Analysis (PCA) to reduce the dimensionality. By averaging the projection of the data onto the first mode for all correct/rewarded trials and all incorrect/unrewarded trials, we find separate phenomena for each trial type (Fig. 8.8C).

For NS-DMD, we average $f(t)$ across correct and incorrect trials for all modes that occur within 27 and 37 Hz. We find the same amplitude trends in Fig. 8.8D, where a large amplitude occurs in incorrect trials one second after feedback. The scale is different in Figs. 8.8C and 8.8D due to normalization. The similarity between overall trends indicates that NS-DMD recovers similar results to standard Hilbert analyses.

Next, we compare NS-DMD to the phase of the Hilbert transform. Jutras et al. (2013) [43] showed that eye movements are associated with phase resets. We explore whether feedback events also cause a phase reset. We bandpass and Hilbert transform every channel between 2 and 4 Hz. The angle of the Hilbert transform finds the instantaneous phase for every millisecond of every trial. We find a consistent phase shortly after feedback begins. A phase histogram for an example channel is shown in Fig. 8.8E. We compare the phase in the NS-DMD modes, $\angle\phi$, for the same example channel. The histogram of phases of all modes between 2 and 4 Hz is shown in Fig. 8.8F. The same phase reset can be seen in the NS-DMD phases.

Lastly, we compare NS-DMD to an example coherence analysis. We calculate the coherence between every pair of electrodes with the Welch method at 3.5 Hz for 500 ms after feedback begins. We then find the average phase difference between all pairs of channels across correct and incorrect trials, given

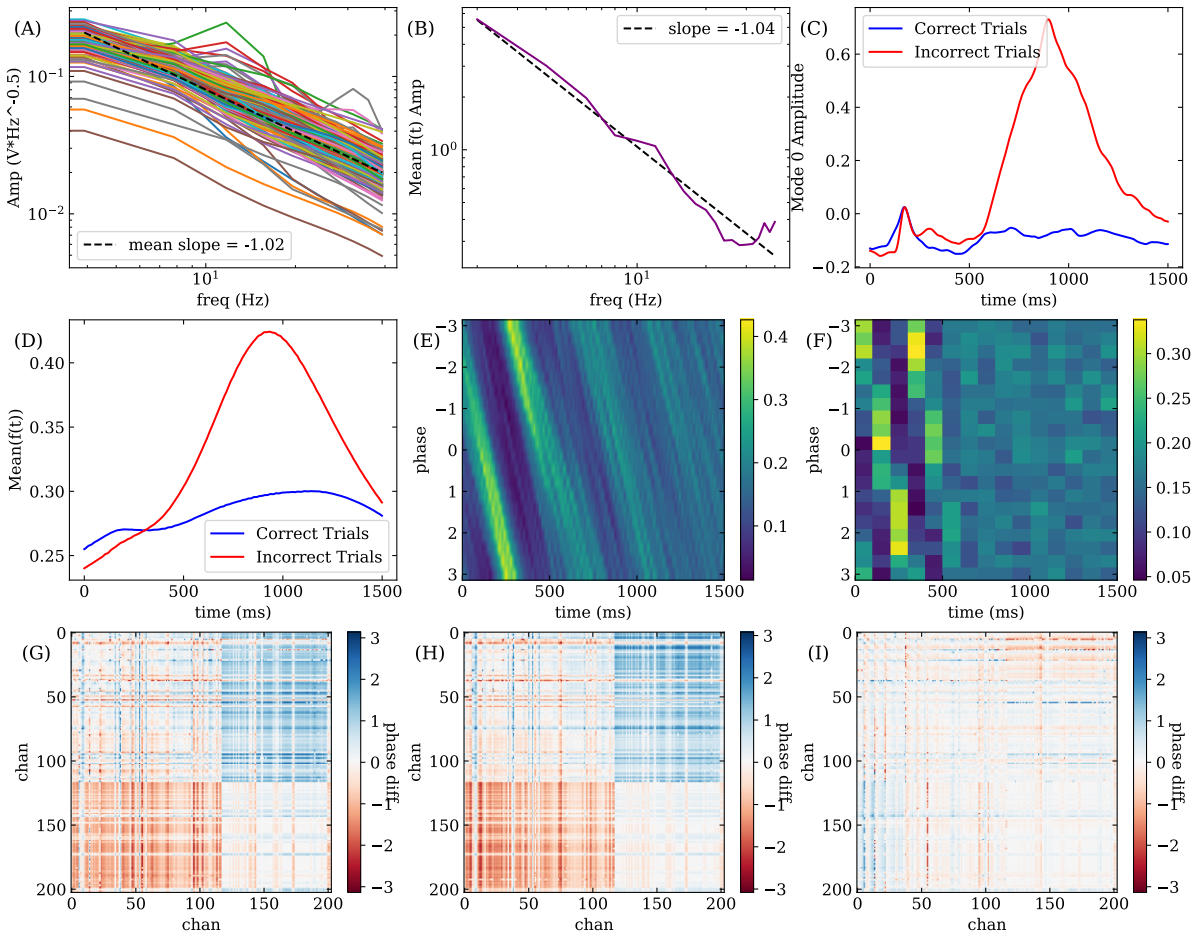

Figure 8.8: Comparisons of NS-DMD results with traditional analyses. (A) shows the Power Spectral Density plots for every electrode along with their mean power law decay. For the NS-DMD modes, (B) shows the average $f(t)$ for every 3 Hz band along with the slope. NS-DMD finds a similar, but mildly different, power law decay to (A). After bandpassing and Hilbert transforming the data between 27 and 37 Hz, (C) shows the first PCA mode after feedback begins at $t = 0$ ms. For NS-DMD, (D) shows the mean $f(t)$ for all modes between 27 and 37 Hz, averaged across correct and incorrect trials, matching the trend in (C). The difference in scale is due to the normalization in the Hilbert method. (E) shows the normalized histogram of phases, determined by the Hilbert transform between 2-4 Hz, across all trials for each time point. The normalized histogram of the NS-DMD phases between 2-4 Hz is computed across all trials for each time point in (F). To compare with standard coherence, we take the average phase difference between every pair of channels for incorrect trials in (G). Coherence is calculated at 3.5 Hz for a window of 0-500 ms between every pair of channels, and the phase difference is averaged with Eq. 8.18. All average phase values are set to 0 when the amplitude of the average phase vector is less than 0.1. (H) shows the average phase difference between every pair of channels for incorrect trials, computed with NS-DMD. Modes are averaged, where the modes are between 2-4 Hz and where $\mathbf{f} > \bar{\mathbf{f}}$ within 100-300 ms. NS-DMD finds a similar answer to (G) with a mean squared error of 0.17. The difference is shown in (I).

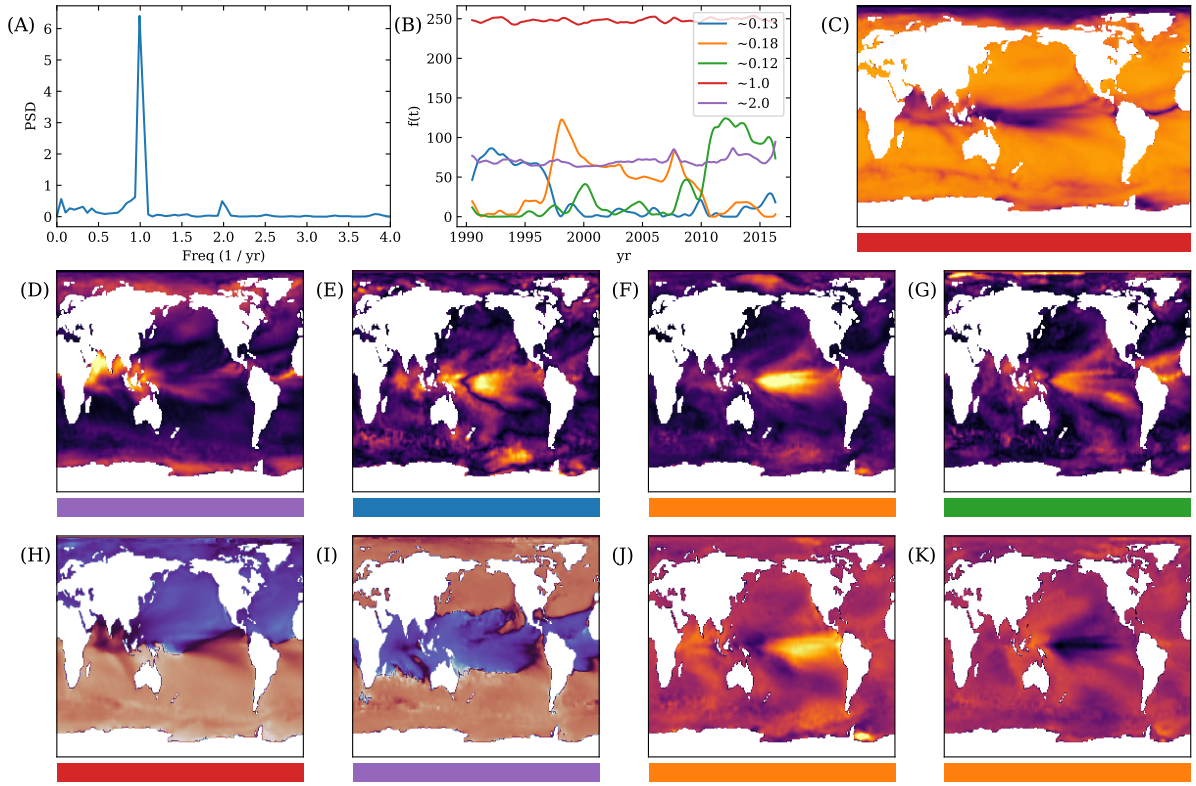

Figure 8.9: Application of NS-DMD on Sea Surface Temperature data ranging from 1990 to 2016. (A) shows the power spectrum density for a random location in the Pacific Ocean. (B) shows the amplitudes of the 5 resulting NS-DMD modes **F**. The legend highlights the average frequencies for each mode in units of cycles per year. The two stationary modes occur due to seasonal effects. The other three modes tile the duration of the dataset and are most likely El Niño and La Niña modes. For (C) through (K), the horizontal bar showcases which mode it comes from, in terms of the colors in (B). (C) shows the spatial amplitude $|\phi|$ for the once a year mode. The relative phase $\angle\phi$ is shown in (H), where the Northern and Southern hemispheres are a π phase apart. The spatial amplitude for the twice a year mode is shown in (D) and the relative phase is shown in (I). The phase is π offset across the tropic of Cancer and Capricorn latitudes. (E-G) show the spatial amplitudes for the El Niño and La Niña modes, where the strength is particularly strong in the Pacific ocean. The data is reconstructed from the orange mode for a week in January in 1998 (J) and 1999 (K), which reconstruct the very strong El Niño event in 1998 and the very strong La Niña event in 1999.

by Eq. 8.18. We consider only phases for which the magnitude of the average phase vector $\overline{\theta} > 0.1$. The resulting spatial phase difference map for incorrect trials is shown in Fig. 8.8G. For NS-DMD. We find the phase difference between all pairs of electrodes for modes between 2 and 4 Hz and when $f(t) > \overline{f(t)}$ between 100 and 300 ms. We average the phase differences with Eq. 8.18. The resulting spatial map in Fig. 8.8H matches the coherence analysis. The difference is shown in Fig. 8.8I, highlighting some small, local differences. The mean squared error (MSE) is 0.17.

8.4.3 Application to Sea Surface Temperature

We apply NS-DMD to sea surface temperature (SST) data, where known global frequencies exist. SST data (NOAA Optimum Interpolation (OI) Sea Surface Temperature (SST) V2 [227]) is collected via satellite, and it is averaged weekly from 1990 to 2016 on a 180 by 360 grid across all longitudes and

latitudes. After flattening and removing land locations, we end with a length 44219 vector for 1455 weeks. The data for each recording location are normalized by z-scoring across weeks.

The power spectrum density (PSD) of a sample location in the Pacific Ocean is shown in Fig. 8.9(A), where there appears to be different frequencies: there is a high amount of power once per year, a smaller amount of power twice per year, and a fluctuating amount of small power less than once per year.

From the PSD, we estimate that 6 modes per window are sufficient to reconstruct the data. We run NS-DMD with 6 modes per window, a window size of 150 weeks, a stride of 25 weeks, and we guess the 6 modes have frequencies of $\pm$ 1, $\pm$ 2, and $\pm$ 0.12 per year. We compute for the first 1400 weeks. Specific parameters are labeled in App. 8.6.2. We find that 5 modes, that span the entire 1400 weeks, reconstructs the data well. The cosine distance between the reconstruction and original, z-scored data is about 0.92.

The amplitudes $\mathbf{F}$ are shown in Fig. 8.9(B) for the 5 modes. There are two modes that exist with a constant amplitude at once and twice per year. These modes correspond to seasonal changes in the SST. The once a year mode is present across the entire ocean (Fig. 8.9(C)). Examining the phase in Fig. 8.9(H) shows a π offset in the phase from the Northern and Southern hemispheres, occurring due to the tilt of the Earth. The twice a year mode occurs in equilateral locations with an emphasise in the Indian Ocean (Fig. 8.9(D)). This also occurs due to the seasonal tilt of the Earth, where equilateral locations undergo a twice a year heating event when the Sun shines most directly on it. The phase of the twice a year mode is shown in Fig. 8.9(I), where we see a π offset between the equilateral and non-equilateral locations around the Tropics of Cancer and Capricorn.

The other three modes in Fig. 8.9(B) are El Niño and La Niña modes, and occur with frequencies of 0.12, 0.13, and 0.18 per year. El Niño and La Niña are typically referenced to occur about once every 6 years, or with a frequency of 0.167. These three modes span the 1400 weeks, where when one mode turns off, another turns on. The spatial amplitude is shown in Figs. 8.9(E)-(G), where one can see the characterizing large amplitude in the equilateral latitudes in the Pacific ocean. To confirm, we analyze the orange, 0.18 times a year mode in 1998 and 1999, which were known as exceptionally strong El Niño year and La Niña years. By looking at the orange mode during a week in January of 1998 and January of 1999, we see a relative increase and decrease in the temperature near the equator in the Pacific Ocean (Figs. 8.9)(J) and (K).

We highlight that NS-DMD is applicable to sea surface temperature data, and it can find modes of particular interest. It is appropriate for this problem due to the combined stationary and non-stationary modes. The seasonal changes in temperature are stationary, but the El Niño and La Niña effects are non-stationary.

Other groups have approached SST data with DMD like methods: [190] uses multi-resolution DMD, [197] uses Time-varying Autoregression with Low Rank Tensors (TVART), and [188] uses BOP-DMD to

analyze SST data. They have all shown success in finding modes correlating with El Niño and La Niña in the Pacific Ocean. Our approach, however, allows us to find differences in the spatial amplitude during different years. The modes attributed with El Niño and La Niña turn on and off during two specific years, indicating that the structure may be changing slightly.

8.5 Discussion

We introduce a novel method for analyzing time-series data: Non-Stationary Dynamic Mode Decomposition. NS-DMD builds on previous DMD methods by including global modulation and time dependent modes. Thus, any improvement to standard DMD algorithms can be easily integrated into NS-DMD. NS-DMD accurately discovers modes that well explain data across a range of simulated settings (Sec. 8.4.1). Naturally, this method is best suited for data that includes low-rank spectral features. It is possible to run DMD on any time-series data, since any signal can be decomposed into a Fourier series basis set of sines and cosines.

NS-DMD can be useful in many empirical settings. This is because many systems elicit non-stationary behavior. For such systems, NS-DMD is better suited than previously proposed methods that assume stationary properties. In the present work we demonstrated this in data from large-scale neural recordings and from recordings of SST. In the neural data, NS-DMD recovers a plethora of modes that capture both short and long time-scale dynamics. Further, it captures widespread activity and information flow between different brain areas. In the SST data, NS-DMD is able to capture consistent global temperature phenomena, as well as subtle differences in the El Niño effects across the years. For these empirical data, NS-DMD extends and subsumes standard methods, such as spectrograms, wavelet transforms, or coherence. However, while these methods work on individual recording locations or can be combined with global dimensionality reduction techniques, the main benefit of NS-DMD is to simultaneously gather spatial information, spectral information, and growth and decay of all modes.

NS-DMD can capture drifting components since it allows for modes to modulate slightly over time. It is then useful to combine modes into one single drifting mode instead of defining multiple new processes. Among its limitations is the fact that NS-DMD requires careful choice of the correct values of hyperparameters. If the similarity threshold is too tight, a single mode will be parsed into many chunks at different short intervals of time. If the similarity threshold is too broad, we lose the ability to distinguish between mode switching and time dependent modes. Expert knowledge can help with hyperparameter choice. Alternatively, to decide on hyperparameters, we recommend directly analyzing modes from consecutive windows. E.g., if the modes gradually change, the similarity hyperparameters should allow these modes to be defined as similar. Or if the modes turn on or off rapidly, a tighter similarity hyperparameter can be used. To decide on the OPT-DMD rank per window, one can observe

the number of power spectral density (PSD) peaks, where each peak will likely be associated with two OPT-DMD modes. The PSD can also be used to initialize the modes. See Sec. 8.4.3 for an example of this approach.

8.5.1 Related Work

Many previous approaches exist to fit non-stationary systems that are closely related to NS-DMD. In this section, we describe some of this previous related work, and draw distinctions between these previous approaches and the NS-DMD approach developed in the present paper. These include hidden Markov models [203, 207, 208], and time-varying autoregressive models [209]. Generally, these methods contrast with our approach in assuming discrete state transitions and do not fully capture the dynamics in terms of identifying spatiotemporal modes of the system. Piece-wise Locally Stationary Oscillation models [204] and state-space multi-taper methods [130] focus on non-stationary estimates for univariate time-series recordings. However, while these approaches independently model individual variables in a multi-variate time-series, they do not find low dimensional spatial modes combining rows/variables.

Variations of DMD [8] seek to address the full spatiotemporal dynamics, including finding reduced dimensionality spatial modes of oscillatory dynamics, across large, multi-variate systems [189, 190]. While powerful in principle, DMD is highly sensitive to noise [187, 205], thus generating biased and inaccurate estimates of the dynamics. Optimized DMD provides the most stable and biased estimate of a DMD model [187], with the bagging, optimized DMD (BOD-DMD) method [188] improving the method even further by providing uncertainty estimates of the DMD fit. But these DMD methods still fail when the generating dynamical system switches between different approximately linear regimes [206]. Within the Koopman framework, Macesic et al. [189] introduce two methods for dealing with rapid switches in the underlying system as well as continuously varying time-series. Although, expert knowledge of the system is needed to introduce observables that linearize the dynamical system. One variation of particular interest, Multi-Resolution DMD [190], specifically accounts for non-stationary time-frequency data. In this approach, one seeks DMD modes at different frequency scales, and with smaller and smaller windows, which increases the temporal resolution. The method successfully identifies non-stationarities, e.g., the El Niño effect in ocean temperature data. One downside, however, is that expert knowledge of the appropriate window size at different scales is required. This also assumes that lower frequency modes are more stable in time. While this may be appropriate for some systems, such as ocean temperature, this is not generally true for all non-stationary systems.

Switching Linear Dynamical Systems (SLDS) [93, 179–181], assumes a Markov process that switches between discrete, linear systems. A recurrent version was developed for neuroscience applications in [181]. These types of models find discrete states and transitions between them. NS-DMD can improve on this by allowing the states themselves to modulate over time, i.e. with a continuously variable amplitude or

frequency. It also allows for independent transitions in individual modes or spatiotemporal components of the dynamics, without requiring an entire state to transition.

Another recent addition to the toolbox of methods for time varying systems is Latent Factor Analysis via Dynamical Systems (LFADS) [178, 182], in which smooth, low-dimensional dynamics are inferred using deep learning based on initial conditions and inferred inputs. While LFADS was first developed for neuronal spike counts, modeled as point processes driven by underlying latent dynamics, recent work [228] has extended LFADS to continuously varying signals. This method does not assume linear dynamics and instead uses recurrent neural networks to find low dimensional factors. Since NS-DMD uncovers linear approximations, some representations may be easier to interpret, such as the leading and lagging of individual spatial areas.

Lastly, a recent approach to non-stationary time-frequency data is Time-Varying Autoregression with Low-Rank Tensors [197], which successfully identified low rank modes and crossover points for constantly evolving data. This method is extremely similar to NS-DMD, even finding global modulations of individual modes. However, instead of finding dynamics in the form of Eq. 8.3, they find the global modulation of two spatial components; there is a lack of frequency and phase of each spatial mode. E.g., if one is interested in the global modulations of modes at a particular frequency or if one is interested in the phases of the spatial distribution, NS-DMD can be more informative.

8.5.2 Future Directions

There are a number of potential additions for improving the effectiveness of NS-DMD. First, we implemented Optimized Dynamic Mode Decomposition [187], which is known to be more robust to noise than the standard DMD method. In the future, we plan on adding Bagging Optimized DMD (BOP-DMD) [188] to NS-DMD. In BOP-DMD, one runs OPT-DMD many times for each window to find statistics of each mode. BOP-DMD also can provide a metric to quantify uncertainty as it automatically produces probability density estimates for the modes, eigenvalues and loadings of the DMD approximation. This could aid with some cases where excess, poorly estimated modes are removed.

In the gradient descent method, we have imposed non-negativity and continuity by manually setting negative values to zero and smoothing across time. Given advances in Non-Negative Matrix Factorization (e.g. [229]), we believe it possible to further optimize the gradient descent method to implicitly restrict the values. We also believe we can implicitly add averaging into the methods.

Another assumption of NS-DMD is that the data are real. If the data are instead complex, one can easily transform it by squaring the magnitude. In the future, however, we plan on generalizing the gradient descent method to allow for complex inputs.

In allowing modes to disappear for extended periods of time, one expects in general that they can reappear with a phase unrelated to the previous appearance. However, NS-DMD retains phase informa-

Simulation	Freq. Diff.	$\|\phi\|$ Diff.	α	β	N
Two-Assemblies	0.2	0.92	0.1	0.1	20
Two-Assemblies (small rank)	0.2	0.97	0.1	0.1	20
Two-Assemblies (large rank)	0.2	0.92	0.1	0.1	20
Freq Coupling	0.2	0.98	0.1	0.1	8
Coupling Lag	0.2	0.95	0.1	0.1	3
Freq Power Law	0.2	0.90	0.0	0.0	30
Freq Drift	1.0	0.95	0.07	0.1	20
Spatial Drift	0.2	0.92	0.1	0.1	20
Mode Return	0.2	0.98	0.1	0.1	20
LFP Data	0.5	0.90	0.0	2.0	30
SST Data	2	0.90	0	0	5

Table 8.1: Table of similarity parameters (L), which describes the thresholds for different windows to be considered "similar". Similar modes are grouped together. We omit the ϕ angle similarity threshold, which we set to 10 for all cases. This threshold is compared to the mean squared error. (R) describes the hyperparameters for gradient descent. For the β term and averaging between iterations, we use N consecutive time points.

tion about modes. As we explore in the supplementary simulations (Sec. 8.7.3), under these conditions NS-DMD will either add an entirely new mode or mix modes. Ideally, phase should reset anytime that $\mathbf{F}$ reaches 0. In the future, we plan on implementing this by checking similarity in non-consecutive windows, allowing for merging of modes differing only by phase when delayed by large time intervals. This should remove any mode mixing and help with interpretability.

Lastly, we have implemented relatively simple feature selection algorithms. Given the large body of work in this area, we plan on adding other methods, as implemented and reviewed in [217] and [230].

These additions should help with both speed and accuracy of NS-DMD. In the meantime, the current rendition of NS-DMD works very well in many simulations and on a range of different empirical data, and the method already has the power to elucidate systems that were previously intractable.

8.6 Appendices

8.6.1 Code Availability

Python code implementing NS-DMD and worked examples can be found at `https://github.com/learning-2-learn/nsdmd`. All simulations, including supplementary ones, are included here.

8.6.2 Hyper-Parameters

All hyperparameters are listed for the simulations, LFP data, and SST data. The parameters are listed in the order they appear in NS-DMD, and we reference the steps listed in Sec. 8.3.2.

- Step 1 (optional) we bandpass the data for the frequency decay simulation (Sup. Sec. S2-A) and trim $1500ms$ off the ends. We bandpass the LFP data (Sec. 8.4.2) and trim $500ms$ off the ends.

- Step 2: all simulations use a window size of $500ms$ and a stride of $100ms$ except for the frequency drift simulation (Sec. 8.4.1), where we use a window size of $200ms$ and a stride of $50ms$. All simulations use an OPT-DMD rank of 4 except for the four assembly simulation (Sec. 8.4.1), where we use 2, 4, and 6.
- Step 3:, the similarity thresholds are listed in Table 8.1.
- Step 4 (optional): when using an OPT-DMD rank of 6 in the four assembly simulation, we require two or more consecutive similar modes in a row. In the LFP and SST data, we require three or more consecutive similar windows in a row.
- Step 5: for all simulations, we smooth the frequency in $\mathbf{S}$ with a moving average of size $51ms$.
- Step 5 (optional): we include a channel specific temporal lag in for the simulation in Sup. Sec. S2-B.
- Step 6 (optional): for the frequency decay simulation (Sup. Sec. S2-A), we remove all individual modes where the reconstruction error is less than 0.2.
- Step 7 (optional): In all simulations except those listed below, we feature select using the exact method (Sup. Sec. SA-1), and we use a variance threshold of 0.01. In the cases of using a large rank in Sec. 8.4.1 or in the frequency die off simulation (Sup. Sec. S2-A), we use gradient descent with a maximum number of iterations of 5. In all simulations, we run the SBS feature selection algorithm and terminate at 1 mode. In the frequency decay simulation, (Sup. Sec. S2-A), we use the SFS algorithm and terminate at 30 modes.
- Step 8: the parameters for gradient descent are described Table 8.1. In all simulations, we use a maximum iteration of 100, a learning rate of 0.01, a momentum of 0.9, and a low pass filter of the initial guess at $2Hz$.

8.7 Supplementary Material

In the supplemental materials, we add two extra approaches one can take with NS-DMD. These approaches aren't recommended due to additional noise or lack of additional information, but they are included for completion. We also add a variety of simulations for interesting edge cases.

8.7.1 Exact Method to Bypass Gradient Descent

By looking at NS-DMD, one could imagine directly solving for the modulation $\mathbf{F}$ instead of performing gradient descent. This approach tends to include noise in the result, but it is included here for completion.

It can speed up some computations, so it can be useful during feature selection.

A direct approximation of $\mathbf{F}$ begins with the following:

$$\mathbf{x}_j \approx \sum_{\hat{k}} \hat{\mathbf{s}}_{:\hat{k}j} \hat{f}_{\hat{k}j} = \sum_k \mathbf{s}_{:kj} f_{kj} + \mathbf{n}_j, \tag{8.19}$$

where $\mathbf{n}_j \in \mathbb{R}^N$ is a new term representing the noise. Hats $\hat{\cdot}$ indicate approximations of the true modes. Any true mode k in $\boldsymbol{\mathcal{S}}$ that we do not find in the modes $\hat{k}$ of $\hat{\boldsymbol{\mathcal{S}}}$ is included in $\mathbf{n}$. For each snapshot t_j and mode k, we take the gradient of the L2 norm (Eq. 9 in the main text) and set $\hat{f}_{k' \neq \hat{k},j} \equiv 0$ to get:

$$\hat{f}_{\hat{k}j} = \frac{\mathbf{x}_j \hat{\mathbf{s}}_{:\hat{k}j}}{|\hat{\mathbf{s}}_{:\hat{k}j}|^2}. \tag{8.20}$$

This effectively finds individual $\hat{\mathbf{F}}$s for each mode. Since our goal is to estimate $\mathbf{F}$, not $\hat{\mathbf{F}}$, we substitute the true solutions for $\mathbf{X}$

$$\hat{f}_{\hat{k}j} = \frac{(\mathbf{S}_j \mathbf{f}_j + \mathbf{n}_j)\hat{\mathbf{s}}_{:\hat{k}j}}{|\hat{\mathbf{s}}_{:\hat{k}j}|^2}, \tag{8.21}$$

and define the following terms:

$$b_{k\hat{k}j} \equiv \frac{\mathbf{s}_{:kj}\hat{\mathbf{s}}_{:kj}}{|\hat{\mathbf{s}}_{:\hat{k}j}|^2} \quad \text{and} \quad \tilde{n}_{\hat{k}j} \equiv \frac{\mathbf{n}_j \hat{\mathbf{s}}_{:\hat{k}j}}{|\hat{\mathbf{s}}_{:\hat{k}j}|^2}. \tag{8.22}$$

Our equation reduces to

$$\hat{\mathbf{f}}_j = \mathbf{f}_j \mathbf{B}_j + \tilde{\mathbf{n}}_j. \tag{8.23}$$

The next few steps serve to invert the above equation to find $\mathbf{F}$.

For the inversion, we approximate $\hat{\boldsymbol{\mathcal{S}}} \approx \boldsymbol{\mathcal{S}}$ and remove $\mathbf{n}_j$. In other words, we assume that we have captured all relevant modes in $\hat{\boldsymbol{\mathcal{S}}}$, and we ignore any additional modes.

$$\hat{\mathbf{f}}_j \approx \mathbf{B}_j \mathbf{f}_j, \tag{8.24}$$

where the rank of $\mathbf{B}_j$ is less than its number of rows/columns. We use an SVD [10] approach here:

$$\mathbf{U}_j \tilde{\mathbf{B}}_j \mathbf{U}_j^\dagger \mathbf{f}_j = \hat{\mathbf{f}}_j, \tag{8.25}$$

where $\tilde{\mathbf{B}}_j$ is diagonal and $\mathbf{U}_j$ is a unitary matrix that diagonalizes $\mathbf{B}_j$. We reduce $\tilde{\mathbf{B}}_j$ to a low rank r by only including large variance terms, since $\tilde{\mathbf{B}}_j$ is linearly dependent. We use a variance threshold of 0.01. We invert and find an estimate of $\mathbf{F}$:

$$\mathbf{f}_j \approx \mathbf{U}_j \tilde{\mathbf{B}}_j^{-1} \mathbf{U}_j^\dagger \hat{\mathbf{f}}_j. \tag{8.26}$$

The steps to find $\mathbf{F}$ are then to calculate $\hat{\mathbf{f}}$ in Eq. 8.20, $\boldsymbol{\mathcal{B}}$ in Eq. 8.22, and find the SVD of $\boldsymbol{\mathcal{B}}$.

8.7.2 Individual Global Modulation per Channel

It may also be possible that the time dependent functions $\mathbf{f}(t)$ are different for each recording channel. Specifically, we analyze systems such that $x_i(t) \sim x_i(t + \Delta t_{ij})$, where the time dependence of channels i and j are offset by Δt_{ij}. In these systems, both the phase of the spatiotemporal modes $\boldsymbol{\mathcal{S}}$ and the modulations $\mathbf{F}$ are related by the same temporal offset Δt_i. We can define the time dependent function $\mathbf{g}_i(t) \equiv f(t + \Delta t_i)$ to distinguish the time dependence of every channel from the global modulation $f(t)$. The spatiotemporal modes are the same as before: $\mathbf{S}(t) \equiv \mathbf{\Phi}(t)e^{i\angle\mathbf{\Lambda}(t)t}$, where the phase of each channel $\angle\mathbf{\Phi}_i = \angle\mathbf{\Lambda}(t)\Delta t_i$. We then have:

$$\mathbf{x}_i = \sum_k \mathbf{s}_{:ki}\mathbf{g}_{:ki}. \tag{8.27}$$

To calculate the temporal lag, Δt_i, one can use $\angle\phi$. However, this only works when the lag is less than half of a period; when any larger, the total time delay is ambiguous due to periodicity. A more general method is to instead check the reconstruction error for all possible temporal lags. Each lag can be compared to knowledge of the system: e.g delays between measurements in fluid flow or expected transit times between neurons in the brain. We expect the lag to be approximately a 2π multiple of the phase of ϕ. However, without knowledge of the system, we have not found a general way to automate finding the exact temporal lag.

From simulations in Sec. 8.7.3, we find that this additional feature does not help improve the fit enough to warrant its use. It is included as an optional step.

8.7.3 Additional Simulations

We now apply NS-DMD to more simulated data, in situations that either require optional steps in the NS-DMD method or are specifically interesting to some communities. We simulate synthetic data with the following generic equation:

$$\mathbf{x}(t_i) = \sum_k f_k(t_i)\mathbf{g}_k(t_i + \frac{\phi_{P,k}(t_i)}{\omega_k(t_i)})\phi_{A,k}(t_i) * cos(\sum_{j=0}^{i-1}\omega_k(t_j)\Delta t + \phi_{P,k}(t_i)), \tag{8.28}$$

Power Law Spectral Distribution

An important issue is having a system composed of many different frequencies at different magnitudes of power. One example, that occurs in many communities, such as physics biology, economics, etc. [231], are frequency power laws. It is common for signal to occur where the power fluctuates like $1/f^{\alpha}$. This case requires the optional step of bandpassing the data into smaller bands to find modes with smaller

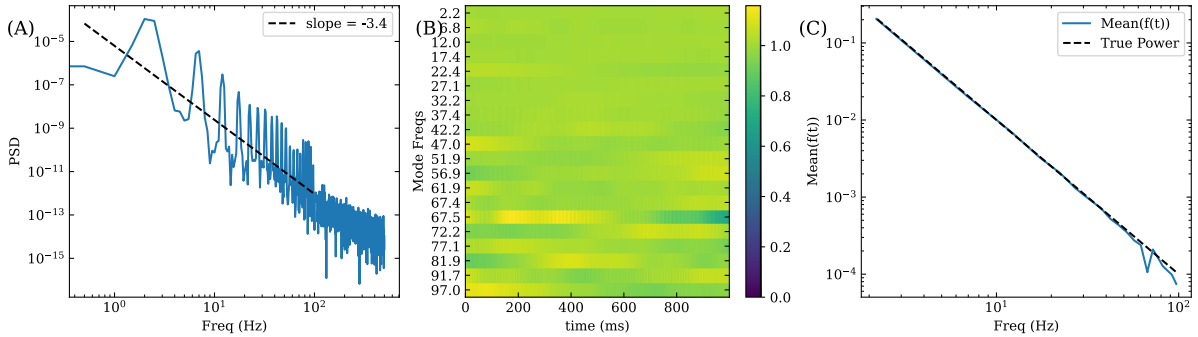

Figure 8.10: (A) shows the Power Spectral Density (PSD) of the first channel. The peaks indicate the frequencies of each mode, which have a power law decay of -2. The frequencies occur at $2 + 5nHz$. The background noise is added at a slope of -2.5. All channels have a similar PSD. (B) shows $f(t)$ for each mode, normalized by their mean. NS-DMD approximately recovers the true, flat distribution equal to 1. However, there is an extra $67Hz$ mode and no $87Hz$ mode. (C) shows the true power law decay (dotted line) compared to the mean of $f(t)$ of each mode. NS-DMD recovers the true power law.

amplitudes.

We simulate power law decay with random spatial modes $\mathbf{\Phi}$ that occur in $5Hz$ increments from $2Hz$ to $100Hz$, where the amplitude occurs with an exponent of $\alpha = -2$. For simplicity, we do not include any non-stationarity in the amplitude, which corresponds to the following simplification of Eq. 8.28:

$$\mathbf{x}(t_i) = \phi_{A,k} cos(\omega_k t_i + \phi_{P,k}). \tag{8.29}$$

Random noise is added to each frequency with a standard deviation of 0.5. Fig. 8.10A shows the power spectrum of an example channel. Due to pink noise, we find an exponent of -3.4 when fitting a line in log space. Modes become blended together at higher frequencies. I.e. for a $1s$ increment, 2 and $3Hz$ are very different while 92 and $93Hz$ are very similar. Thus, this simulation tests NS-DMD at the ability to segregate small and large frequency differences.

The result of NS-DMD is a relatively flat spectrum of $\mathbf{F}$ (Fig. 8.10B), exactly under the simulation parameters. Note that we normalize $\mathbf{F}$ for every mode for visualization. We show the mean of $\mathbf{F}$ for all frequencies in Fig. 8.10C, where we see that the amplitude decays like the true power law.

NS-DMD doesn't perform perfectly at higher frequencies, where in this example there is an extra $67Hz$ mode and lack of $87Hz$ mode. NS-DMD tends to perform better at lower frequencies where the modes are better segregated from each other.

Frequency Coupling and Independent Channel Modulation

We turn our attention to a neuroscience issue, where it's believed that low frequencies modulate high frequencies via frequency coupling [39, 221, 223, 232]. The lower frequency signal likely has a signal with an unknown exact form. Thus, we simulate a square wave, where all possible frequencies are part of the low frequency signal. The square wave switches between 0 and 1 with a periodic frequency of $5Hz$

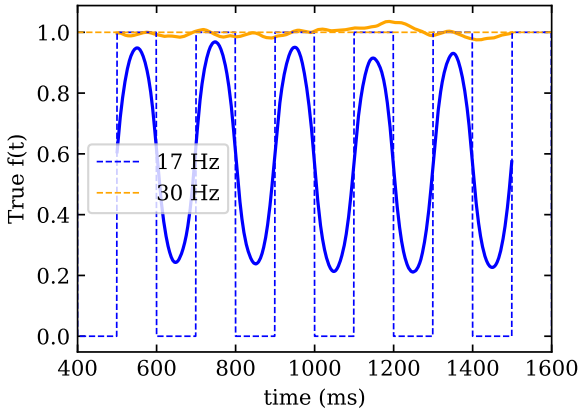

Figure 8.11: Comparison of the true (dotted lines) and fitted $f(t)$ (solid lines). NS-DMD is able to find the dominate, $5Hz$ Fourier mode of the this coupling. Depending on the amount of smoothing, one can be more or less accurate during the steepest portions of the square wave. Success with NS-DMD on square wave coupling demonstrates the ability to work on any periodic coupling. This is due to the Fourier series of a square wave including all frequencies.

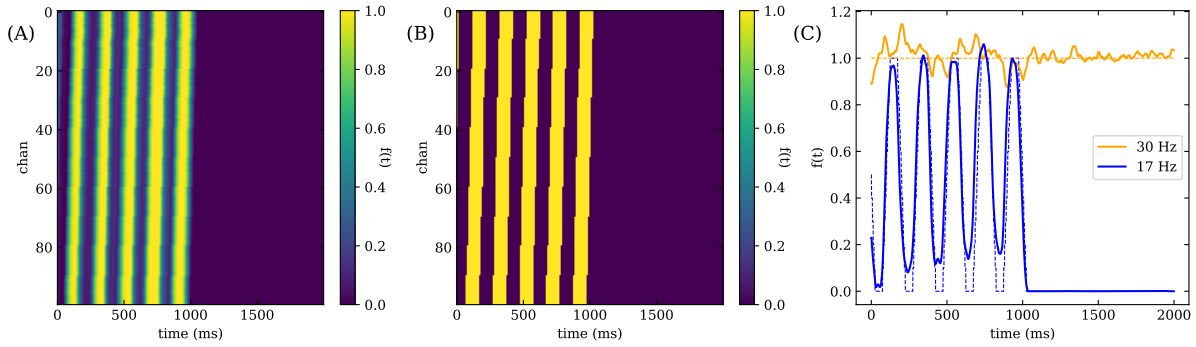

Figure 8.12: We show the results of independent modulations for each channel at $17Hz$. We compare the fit $f(t)$ for each channel in (A) to the true $f(t)$ in (B), where we find that NS-DMD recovers the ground truth. Note that this only works when the overall lag is less than a period at the modes frequency. We also show in (C) a comparison of the true $f(t)$ (dotted lines) and fitted $f(t)$ (solid lines) when we do not include a temporal lag for **F**. NS-DMD still recovers the correct overall global modulation. The true $17Hz$ modulation is trapezoidal, which is due to averaging a square wave with different temporal lags.

(Fig. 8.11). Using NS-DMD, we find the dominating $5Hz$ frequency in Fig. 8.11. Since NS-DMD works with a square wave, it likely works with any periodic function.

We modify this simulation to explore the case when individual channels are independently modulated, as developed in Sec. 8.7.2. For this example, each channel has the same modulation, offset by a temporal phase, which corresponds to the following simplification of Eq. 8.28:

$$\mathbf{x}(t_i) = \mathbf{g}_k(t_i + \frac{\phi_{P,k}}{\omega_k})\phi_{A,k}cos(\omega_k t_i + \phi_{P,k}). \tag{8.30}$$

We generate the data with two assemblies. The first assembly has a constant amplitude at $30Hz$. For the second assembly, each channel is modulated by a square wave that stops around $t = 1000ms$ (Fig. 8.12B).

While running NS-DMD, we implement the optional step of including temporal lags for each channel based on the angle of $\mathbf{\Phi}$. In Figs. 8.12A and 8.12B, we show that NS-DMD recovers the correct global modulation $\boldsymbol{\mathcal{G}}$ for each individual channel. We also ran NS-DMD without the optional temporal lags, where NS-DMD finds an approximation of $\boldsymbol{\mathcal{G}}$ (Fig. 8.12C). With this in mind, along with the requirement that the temporal delay is less than half a period, the optional inclusion of temporal lags in NS-DMD may not be needed.

Modes Returning With New Phase

When $\mathbf{F} = 0$, the phase is undefined, and it's possible that an assembly will turn off and back on with a different phase. If the phase difference is small, we expect NS-DMD to find one mode that explains the assembly before and after $\mathbf{F} = 0$. If the phase difference is large, up to π, we expect NS-DMD to find two modes that explain the data. To analyze the behavior of NS-DMD on this effect, we generate synthetic data from two assemblies. The first mode has a constant $f(t)$ for the duration of the dataset at $12Hz$ with a randomly generated spatial mode ϕ. The second assembly turns off at $\sim 500ms$ and back on at $\sim 1000ms$ with a frequency of $5Hz$. The shape of $f(t)$ is shown in Fig. 8.13. The spatial mode ϕ is randomly generated, and the phase $\angle\phi$ of each channel is advanced 0, $\pi/2$, and π radians for Figs. 8.13A,D, Figs. 8.13B,E, and Figs. 8.13C,F respectively when the mode returns at $\sim 1000ms$.

To analyze the behavior of NS-DMD, we run NS-DMD on each case with a final subset of either two Figs. 8.13A-C or three modes Figs. 8.13D-F. As expected, when the phase does not change, we find two modes is sufficient to capture both assemblies Fig. 8.13A. If instead, we use three modes, we find a mixture of two modes to explain the leaving and returning assembly Fig. 8.13D.

In the other simple case where the phase is advanced π radians, we find that two modes only recovers the assembly at the beginning of the time-series (Fig. 8.13C). Three modes fully recovers the data (Fig. 8.13F).

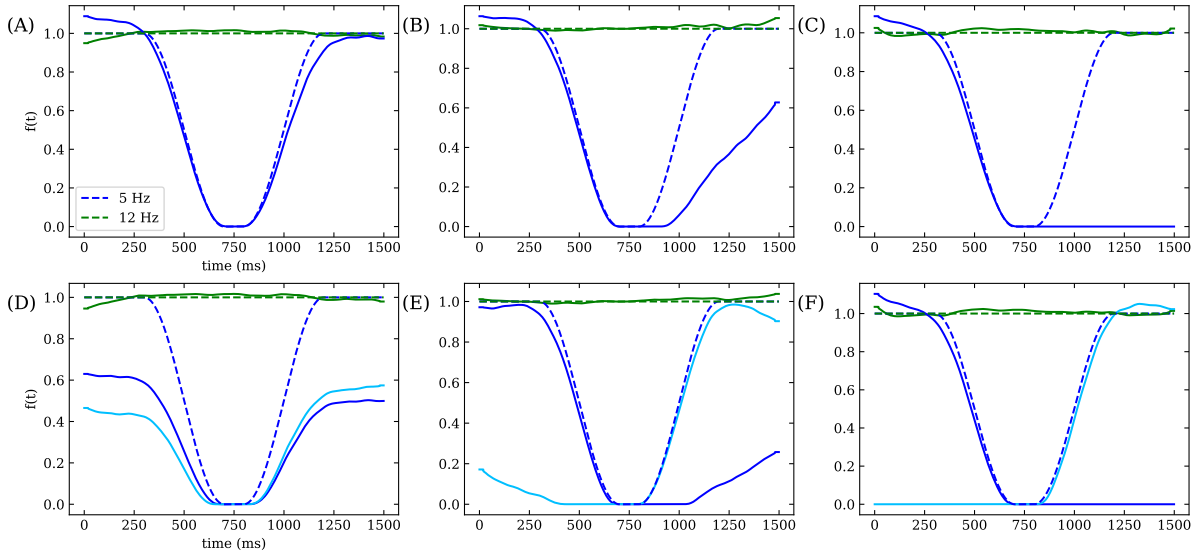

Figure 8.13: This figure shows the true $f(t)$ (dotted lines) and approximated $f(t)$ (solid lines) for different cases of the returning assembly simulation. There are two assemblies, one with a constant amplitude at $12Hz$, and one that turns off and back on at $5Hz$. The spatial mode ϕ is random for each assembly, and the overall phase of ϕ is advanced 0, $\pi/2$, and π radians for (A)/(D), (B)/(E), and (C)/(F) respectively. In (A)-(C), we converge to two modes with NS-DMD. In (D)-(F), we converge to three modes with NS-DMD. When the phase doesn't change, we find that two modes best fit the data (A). When the phase advances π radians, we find that three modes best fit the data. Anywhere in between, we find that two modes underfits the data, and three modes can show a mixing effect.

We see a mixture of effects in the case where we advance the phase $\pi/2$ radians. When converging to two modes, NS-DMD partially recovers the assembly when it returns. With three modes, NS-DMD recovers $\mathbf{F}$, and two of the modes mix a little bit.

8.8 Further Work

8.8.1 Approximation of Modes From Known Systems

I derive how one can connect the modes found with NS-DMD to a known time dependent dynamical system. In the simple case, where one knows the matrices $\mathbf{X}$ in Eq. 8.5, it's straightforward to expand the k modes into the modes found with NS-DMD. The general case, where $\mathbf{X}$ is unknown, is much more difficult. In this case, data is described at any snapshot $t = t_i$ with $\mathbf{x}_{i+1} = \mathbf{A}_i\mathbf{x}_i$. In this case, only $\mathbf{A}(t)$ is known, and the user is interested in connecting this to the data driven approximation of NS-DMD.

By finding the eigendecomposition at every snapshot, one can rewrite $\mathbf{A}_i$ in a diagonalized basis:

$$\mathbf{A}_i = \mathbf{\Phi}_i\mathbf{\Lambda}_i\mathbf{\Phi}_i^{\dagger}. \tag{8.31}$$

Expanding with the diagonalized bases, the data is reconstructed at any snapshot t_j as

$$\mathbf{x}_j = \mathbf{A}_{j-1}...\mathbf{A}_1\mathbf{A}_0\mathbf{x}_0 = \mathbf{\Phi}_{j-1}\mathbf{\Lambda}_{j-1}\mathbf{\Phi}^\dagger_{j-1}...\mathbf{\Phi}_1\mathbf{\Lambda}_1\mathbf{\Phi}^\dagger_1\mathbf{\Phi}_0\mathbf{\Lambda}_0\mathbf{\Phi}^\dagger_0\mathbf{x}_0. \tag{8.32}$$

In our construction of NS-DMD, we ensure that $\mathbf{\Phi}$ and $\mathrm{Im}(\mathbf{\Lambda})$ vary slowly in time. Formally, the former assumption can be written as $\mathbf{\Phi}^\dagger_{i+1}\mathbf{\Phi}_i = \mathbf{I}+\epsilon\mathbf{\Delta}_i$, where $\mathbf{I}$ is the identity matrix, $\mathbf{\Delta}$ is a matrix describing the change of $\mathbf{\Phi}$, and ϵ is a small number. With this, one can commute the matrices and truncate to a small order $\mathcal{O}(\epsilon)$:

$$\mathbf{x}_j = \left[\mathbf{\Phi}_{j-1} + \epsilon\mathbf{\Phi}_{j-1}\left(\sum_{i=1}^{j-1}\mathbf{\Lambda}_{j-1}...\mathbf{\Lambda}_i\mathbf{\Delta}_{i-1}\mathbf{\Lambda}_i^{-1}...\mathbf{\Lambda}_{j-1}^{-1}\right)\right] * \mathbf{\Lambda}_{j-1}...\mathbf{\Lambda}_0\mathbf{\Phi}^\dagger_0\mathbf{x}_0 + \mathcal{O}(\epsilon^2). \tag{8.33}$$

The ϵ term is a nontrivial correction term to the reconstruction of $\mathbf{X}$, where every change $\mathbf{\Delta}_i$ is evolved via the eigenvalues $\mathbf{\Lambda}$ and inverse of the eigenvalues $\mathbf{\Lambda}^{-1}$ defined by $\mathbf{\Lambda}\mathbf{\Lambda}^{-1} = \mathbf{I}$. One is usually most interested in the dominating modes where the variation in $\mathbf{\Phi}$ is small, so all ϵ^2 terms are dropped. One can define the projection onto the initial eigenvector $\mathbf{y}_0 \equiv \mathbf{\Phi}^\dagger_0\mathbf{x}_0 \in \mathbb{R}^p$ and the bracketed term $[\cdot]$ as $\mathbf{\Phi}'_{j-1}$ to get

$$\mathbf{x}_j \approx \mathbf{\Phi}'_{j-1}\mathbf{\Lambda}_{j-1}...\mathbf{\Lambda}_1\mathbf{\Lambda}_0\mathbf{y}_0. \tag{8.34}$$

One can now extract the absolute value of $\mathbf{\Lambda}_i$ to define a purely real time dependent function $\mathbf{f}_j = \prod_{i=0}^{j-1}|\mathbf{\Lambda}_i|\mathbf{y}_0$ to obtain:

$$\mathbf{x}_j = \mathbf{\Phi}'_{j-1}e^{i\sum_{i=0}^{j-1}\angle\mathbf{\Lambda}_i}\mathbf{f}_j. \tag{8.35}$$

Finally, one can construct purely oscillatory $\mathbf{S}_j \equiv \mathbf{\Phi}'_{j-1}e^{i\sum_{i=0}^{j-1}\angle\mathbf{\Lambda}_i}$ to arrive at

$$\mathbf{x}_j = \sum_k \mathbf{s}_{:kj}f_{kj}, \tag{8.36}$$

where k sums over the DMD modes up to some true number of modes. Using this, one can compare the expected spatiotemporal modes $\boldsymbol{\mathcal{S}}$ and modulations $\mathbf{F}$ with the their data driven estimates.

Chapter 9

Discussion

This thesis focuses on behavior and neural activity of subjects performing a variant of the Wisconsin Card Sorting Task, a complex decision making task involving ambiguous credit assignment of a large number of features. There are several key findings. First, through behavioral modeling, monkeys tend to follow a reinforcement learning framework while simultaneously focusing on previously correct or incorrect stimuli. Second, leveraging predictions from the behavioral model, one finds decoding of the subjects "belief" during the feedback period, where the belief is the most likely feature to be chosen from a stimulus based on the model. Third, primarily in low gamma, the hippocampus shows timing differences depending on correct or incorrect feedback while the prefrontal cortex (PFC) shows amplitude differences, suggesting different processes in both areas. Fourth, there are two timescales of interest, before and after ~ 400 ms after feedback, where modes show major differences. These findings demonstrate the importance of the feedback period in updating belief states related to solving this complicated task.

All the studies in this thesis play a role in describing how subjects perform this task. The eye movement and phase reset analysis showcases decision making during the selection period. Demixed principal component analysis showcases stimulus working memory that lasts early in the feedback period. Pure behavioral analysis highlights possible motifs of the subject, providing further evidence for the behavioral models. The remainder of this thesis is to highlight results, further interpretations, and some of the many possible future analyses on this dataset.

9.1 Findings and Interpretations

Use of a Schema

From the behavioral modeling in Chapter 4, evidence points towards monkeys solving the task with a reinforcement learning framework while humans solve the task like a memory limited ideal observer. The

behavioral statistics in Sec. 3.2.3 provide further support for a reinforcement learning framework. So which are used: schemas, mental models, or learning sets?

There are multiple plausible scenarios when learning to solve the WCST. First, one could solve this task with a mental model. That is, after enough time spent learning the structure of the task, one could apply the model that is believed to generate the most amount of reward. This idea is plausible when the model is discrete like the memory limited ideal observer model. E.g. the model would describe all features of the task, where the model updates the "value" per feature to that correlating to positive reward, negative reward, or no information.

Another possibility is that a schema is learned, which associates new information when observed. E.g., a schema would learn that it's better to choose features that were recently rewarded compared to features that were not recently rewarded. This would lead to cognitive models that update values with a preference towards or away from recently rewarded or unrewarded features; this is reminiscent of the reinforcement learning model. One could imagine a strict schema where only recently rewarded features are retained, which could converge to an ideal observer model.

One final possibility is the use of a learning set, where the subject learns to solve the task quickly when new rules are introduced. One of the hallmarks of learning sets is the convergence towards fast learning after many problems [21]. This may be possible for the ideal observer model, where features are "one-shot learned." This may not be appropriate for the reinforcement learning model though, since this model predicts that learning is gradual.

I believe that schemas are the most plausible, since it naturally explains reinforcement learning frameworks. The learned schema helps the individual gradually focus towards recently rewarded features. The hypothesis is that no schema exists at the beginning of learning, and the schema is developed after many problems. Although, it's unclear as to whether the schema formation idea best explains the ideal observer model in humans. In that case, a mental model may best explain human behavior. However, when reward is probabilistic [94], humans follow a reinforcement learning framework; this is more consistent with schema like learning.

Saccade Strategy Differences

One interesting point is how monkey S uses different saccade strategies to make choices. Particularly, they choose quickly while exploiting and slowly while exploring. This provides a natural way to describe confidence, where states of confidence can lead to different saccade strategies.

Further, the subject tends not to use peripheral vision; although, they may sometimes use peripheral vision during color rules. Instead, during exploitation of criterion trials, the subject observes cards until they find the rule card. This supports the idea that the subject chooses a feature "before" seeing the cards instead of integrating all possible information. Towards the beginning of the rule block when all 4

cards are explored, the subject may instead integrate information.

Interestingly, there was a change in strategy when the timeout period was shortened for monkey S. They switched to a strategy where they typically saccaded once when previously incorrect and exploring, indicating a strategy with fast choices when uncertain. However, once correct while exploring, the subject spends time to make a selection, indicating that they search for these correct features. Even when the timeout period was restored, the subject continued with this fast choice strategy and they received the same reward per time spent. This indicates that different strategies lead to similar rewards and that initial task conditions can lead to different strategies.

Working Memory vs Belief State

In Sec. 5.4, one interesting point is the decoding of the belief state compared to a working memory control. Interestingly the working memory representation lasted $\sim$ 400 ms into the feedback period, which is also shown in 3.6. There is an ability to decode the belief state for $\sim$ 400 ms into the feedback period as well; beliefs are decodable with a different representation after $\sim$ 400 ms. Similarly, there is a change point in the feedback representations at $\sim$ 400 ms, particularly in the LFP (Sec. 6.1) amplitudes and information flow at 4-5 and 10-13 Hz.

To update the belief within the reinforcement learning framework, one needs to assign feedback to the chosen features. Given representations of feedback, stimulus working memory, and beliefs, belief updates may occur during the early feedback period.

There are two general hypotheses for what happens after $\sim$ 400 ms. First, a new representation (e.g. in Fig. 5.5) may occur simply because the working memory dropped out. While plausible, it doesn't explain why there is still a somewhat high representational similarity during the selection period and late feedback period. Instead, it's possible that a new representation occurs with similarities to the selection period for a new computation. While plausible, the role of the late feedback computation is unclear.

From these ideas, the hypothesis of working memory integration is the following. First, working memory lasts into the feedback period. The representation may occur in the hippocampus, prefrontal cortex, claustrum, and anterior cingulate. Immediately after the feedback begins, a computation occurs to update the belief state in a reinforcement learning style. This computation occurs partly in the prefrontal cortex, hippocampus, pre-motor region, and basal ganglia. The working memory drops off, and the representation of belief remains. The belief may transmit to many areas of the brain for efficient computation in future trials. The belief may be represented in the prefrontal cortex, hippocampus, basal ganglia, claustrum, and parietal cortex.

Feedback Integration

So how exactly is feedback integrated? From Chapter 6, the hippocampus and prefrontal cortex (PFC) exhibit timing and amplitude differences after positive and negative feedback depending on the previous trials feedback. This indicates that different computations may occur depending on the subjects confidence. E.g., two correct trials in a row likely leads to high confidence in their belief. This idea naturally falls out of the reinforcement learning framework, where (e.g.) multiple correct trials in a row increases the probability of focusing on one feature and thereby increasing confidence towards that feature.

First note the timing difference in the hippocampus compared to the PFC. When the feedback is correct vs incorrect, the PFC has different amplitude peaks at the same time while the hippocampus has a different timing altogether. This is interesting since the subject receives correct feedback faster than incorrect feedback. That is, it's faster to hear a noise and receive a reward compared to realizing that no noise or reward occurs. With this in mind, the hippocampus may provide a signal when it understands the state of the feedback. One can imagine that, if the two areas are coupled, there is an integration of information from the hippocampus into the PFC that affects the amplitude of the PFC mode.

There is also communication between the two areas. Specifically, there is flow towards the hippocampus in the 4-5 Hz band, particularly in incorrect trials. There is flow towards the PFC in the 20-24 Hz band, particularly during correct trials. The opposing directions align with the communication vs coherence hypothesis [39]. Also, the high frequency band flow coincides with the previous hypothesis of the hippocampus coupling with the PFC primarily during correct trials. That is, the early hippocampal signal may be communicated to the PFC, which drives the PFC amplitude. When the hippocampal signal is late due to incorrect feedback, the signal is either inhibited or it arrives late to the PFC. Both hypotheses may occur and drive the PFC to a different amplitude. One PFC mode has a secondary bump in amplitude ONLY after incorrect feedback while a different PFC mode does not.

There is also a quick response in the PFC during correct trials WHEN the previous trial was incorrect. This is an important time to update beliefs, since the subject is likely in a state of uncertainty. The question remains as to why this signal can occur earlier than the hippocampal signal. While it's possible that the hippocampas still directly communicates with the PFC, it's possible that there are different neural pathways directly to the PFC. It's difficult to unpack this result without further analyzing the communication between the two areas while separating based on feedback/previous feedback conditions.

The amplitude effects of the feedback also vary based on the previous feedback. The hippocampal correct feedback mode is boosted when the previous trial was incorrect while the PFC modes are shrunk when the previous trial was incorrect. From a communication perspective, this would mean that there is an inverse relationship between the two areas. This may follow from a belief update perspective: a large amplitude PFC mode may indicate to maintain a belief while a small amplitude PFC mode may

indicate that the belief should be updated. This would explain why two correct trials in a row, leading to belief maintenance, produce a large amplitude mode in the PFC.

In a similar vain, all modes in the hippocampus and PFC increase their amplitude during incorrect trials when the previous trial was correct. From a communication perspective, this would mean there is a relationship between the hippocampus and PFC. However, the communication may be inhibited during incorrect feedback due to the late timing in the hippocampus. Perhaps then these signals have different roles. Differential circuit mechanisms in correct and incorrect trials may be plausible considering that different belief updates are required.

So how is feedback integrated into the belief state update? The work in Chapter 6 aligns with work from Brincat et al. [8], where they hypothesize that the hippocampus guides neocortical plasticity and the PFC encodes associative learning. That is, if the PFC encodes the belief state, the communication hypothesis from the hippocampus to the PFC helps to explain how the PFC updates the belief state under different feedback conditions. Then, the different amplitudes, depending on previous feedback, could help with different update mechanisms.

These kind of belief updates also align with the reinforcement learning schema. In the behavioral models, there are clear differences when the feedback is correct compared to incorrect. The previous feedback may be encoded in a more subtle way. Sequential feedback changes the probability of choosing a feature less after many trials. E.g., one correct trial increases the probability of choosing feature X a lot while many correct trials increase the probability of choosing feature X very little. With this in mind, the results here naturally encode these probability differences.

Time Delay of Stimulus Encoding

There are time delays between saccading to a stimulus compared to understanding the content of the stimulus. In Sec. 3.6, dPCA modes related to stimulus information typically begins after about a hundred milliseconds. Decoding belief in Sec. 5.4, which is related to stimulus information, also occurs after about a hundred milliseconds. From decoding whether the subject will remain or stay on a card in Sec. 7.3.5, decoding occurs after a micro-saccade within the bounding box of a stimulus. This means that decoding only occurs if the subject maintained fixation on a card for a minimum of one inter-saccade interval: approximately 100 ms. There is even a 100 ms delayed response after the stimuli appear on the screen (Sec. 7.3.2). Clearly, there is a delay in processing of visual information of $\sim$ 100 ms. This aligns with previous results [140], where information takes 100 – 130 ms to reach the PFC from the eye.

Hippocampal Phase Reset

One consistent finding with [43] is the resetting of phase in the hippocampus. Specifically, in Sec. 7.2, the hippocampus resets its theta phase (about 9-12 Hz) after both feedback and fixation to the fixation cross. Further, the hippocampus is shown to reset phase after eye movements (Sec. 7.3.1), and there are large amplitude responses after object onset in both the hippocampus and PFC (Sec. 7.3.2).

As suggested by [170–172], modulation of the phase may be important for memory formation and setting up plasticity. Seeing the phase modulations here suggests similar mechanisms, where the hippocampal reset may be useful for belief computations during feedback. Reset during the cross fixation may help initialize the brain state before any actions are performed. Computations immediately after object appearance are important for initial computations of selection. And lastly, phase reset during eye movements is important for interpretation of incoming features relevant for selecting cards. The results in this thesis support the idea that the phase reset in the hippocampus plays a critical role in the decision making and memory formation during this task.

Bringing it all Together

So how do the neural computations and behavioral findings support a schema? As mentioned earlier, the reinforcement learning model supports a schema representation, where the subject takes advantage of the task structure by biasing choices towards multiple recently rewarded features. There is also differential brain activity for repeated stimuli (Sec. 5.5), a result predicted by the best fit learning model (Chapter 4). This also supports a schema, where the subject learns the importance of the previous card. Specifically, the easiest way to obtain reward is to remember the previous card and choose/avoid this card if it repeats.

Computations are necessary in integrating beliefs with incoming stimuli and feedback. Phase reset in the hippocampus is a natural way to prepare the network for incoming processing of information, such as feedback or novel stimuli after an eye movement. The processing of novel stimulus information after an eye movement likely helps for schema-like computations.

The key aspect, however, is the integration of feedback, working memory, and previous knowledge in updating the belief state. Utilizing a schema, the subject can quickly update the belief state after receiving feedback. Integration of feedback, working memory, and previous knowledge gives rise to updating internal beliefs for future trials. The decoding of belief state here and distinctions of feedback support the reinforcement learning framework. While there is still work to be done in understanding the updates to the neural circuitry, the work here supports the use of schemas in processing information relevant for high dimensional credit assignment tasks.

9.2 Future Work

While there are many results showcasing how subjects solve complex decision making tasks, there are still a plethora of analyses that could be done on this task and dataset. E.g., what further estimates of neural circuitry can be found? What about the representation change after feedback? What else can be gathered from the behavioral models? This section suggests further analyses relevant to this task.

There are many further behavioral modeling analyses in this dataset. Particularly, focusing on behavioral modeling incorporating the order of objects seen. Given the lack of peripheral vision (Sec. 3.3.2), the subject likely needs to integrate information while directly observing the cards. This may lead to further predictions of choice depending on the random cards that are observed before others. The behavioral modeling can also be incorporated into the other session groups for monkey S. The work in Chapter 4 focuses on the first group of sessions. However, there are sessions afterwards with different stimuli or shorter timeout periods. Given that the saccade strategy is different for these sessions (Sec. 3.3.2), there may be different models that explain the subject behavior. Combining analyses on these sessions with observed cards may lead to further interesting behavioral insights.

One of the key questions is understanding how the neural circuitry updates upon receiving feedback. To answer this, one method is to examine the neural activity for a trial based on BOTH the model predictions before and after the current trial. In Chapter 5, the focus was on how the belief state of the current trial integrated with the current trial's feedback. An equally promising idea is to focus on how the current trial's feedback influenced the following trial's choice. Other metrics could also be interesting, such as whether the subject changes or maintains belief state between consecutive trials. These analyses may be key in understanding how the current representation updates upon feedback.

In Chapter 4, behavioral models were fit to both monkeys and humans. However, the neural analysis in this thesis focused on the monkey data. It'd be interesting to see the neural correlations to the behavioral models fit to human data, similar to the analyses in Chapter 5. There are key differences in the models. Namely, the best fit model to monkey behavior has values per trials while the best fit model to human behavior has probabilities of discrete states. Still, similar metrics can be found, such as the most likely belief feature on the chosen card or the probability of choosing particular cards. Comparative analyses on the human data may play a key role in understanding inter-species neural differences.

One application of the behavioral modeling approach is the use of reward prediction errors (RPE), which has been shown to be encoded in dopamine neurons [64, 65, 159, 160]. RPE is particularly relevant in the WCST, since the subject may use RPE to update their belief state in conjunction with the reinforcement learning framework. To calculate RPE, one could find the probability of choosing the subjects choice and subtract this from a reward value of 0 (incorrect) or 1 (correct). This approach naturally avoids all specific model values.

Another application of the behavioral modeling is to rescale the model values. The model values may not be the exact values encoded in the brain, as evidenced in Sec. 5.1. Perhaps the brain instead encodes the probability of each feature. Or, one could look for a non-linear correlations between the neurons with the model values. This may lead to an understanding of the encoded information.

As a thought experiment, I propose a few changes to the task design that may improve statistics for future analyses. First and foremost, using three cards would maintain the same task structure while limiting the number of possible rules to only nine instead of twelve. This increases the number of trials per rule per session. However, it's possible that the subjects strategy may change. In a similar vain, it could be helpful to use pseudo-random rule transitions, where rules are guaranteed to occur at least twice per session. It's desirable to have neural representations of the rule multiple times in the session such that long time scale statistics can be ruled out. One minor change is related to the bounding boxes. Currently, to identify object fixation, square bounding boxes are used for all shapes. It may be beneficial to use areas that barely surround each object, especially since neural representations are most present when the subject is directly focusing on each object (Sec. 7.3.4). Further, objects should NOT overlap at all, a bug that currently unfortunately happens. A more subtle change is to ensure that the chosen card does NOT repeat. From the modeling in Chapter 4, monkeys tend to focus on the previously chosen card. However, in removing the ability for the chosen card to repeat, more trials can be used for finding abstract feature representations. These changes highlight the best improvements to the statistics that can easily be incorporated.

References

Articles covering the presented results

1 J. Ferre, A. Rokem, E. Buffalo, J. Nathan Kutz, and A. Fairhall:
"Non-Stationary Dynamic Mode Decomposition",
IEEE Access, 10.1109/ACCESS.2023.3326412 (2023).

2 J. Ferre, P. Q. Zhang, P.-C. Kuo, M. J. Jutras, B. Ram, J. Lin, R. Knight, E. A. Buffalo, E. Y. Walker, and A. L. Fairhall:
"Strategy Differences in Humans and Non-Human Primates Solving High-Dimensional Credit Assignment Problems",
In Preparation (2023).

Books

1 J. O'Keefe and L. Nadel:
The hippocampus as a cognitive map,
(Clarendon Press, 1978), p. 570.

2 P. Andersen, R. Morris, D. Amaral, T. Bliss, and J. O'Keefe:
The Hippocampus Book,
edited by P. Andersen, R. Morris, D. Amaral, T. Bliss, and J. O'Keefe (Oxford University Press, 2006).

3 D. W. Yip and F. Lui:
Physiology, Motor Cortical,
(2023).

4 P. Dayan and L. Abbott:
Theoretical neuroscience: computational and mathematical modeling of neural systems,
(The MIT Press, 2001).

5 P. Mitra and H. Bokil:
Observed Brain Dynamics,
(Oxford University Press, 2007).

6 M. X. Cohen:
Analyzing Neural Time Series Data,
(The MIT Press, 2014).

7 P. Song:
Correlated Data Analysis: Modeling, Analytics, and Applications,
Springer Series in Statistics (Springer New York, New York, NY, 2007).

8 J. Kutz, S. Brunton, B. Brunton, and J. Proctor:
Dynamic Mode Decomposition: Data-Driven Modeling of Complex Systems,
(2016).

9 György Buzsáki:
Rhythms of the Brain,
(Oxford University Press, 2006).

10 S. L. Brunton and J. N. Kutz:
Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control,
1st (Cambridge University Press, 2019).

Articles, proceedings and theses

1 M. Minsky:
"Steps toward Artificial Intelligence",
Proceedings of the IRE **49**, 8–30 (1961).

2 A. Gilboa and H. Marlatte:
Neurobiology of Schemas and Schema-Mediated Memory,
2017.

3 P. Baraduc, J. R. Duhamel, and S. Wirth:
"Schema cells in the macaque hippocampus",
Science **363**, 635–639 (2019).

4 M. K. Benna and S. Fusi:
"Are place cells just memory cells? Memory compression leads to spatial tuning and history dependence",
bioRxiv, 624239 (2019).

5 C. Drieu and M. Zugaro:
Hippocampal sequences during exploration: Mechanisms and functions,
2019.

6 W. Sun, J. Winnubst, M. Natrajan, C. Lai, K. Kajikawa, M. Michaelos, R. Gattoni, C. Stringer, D. Flickinger, J. E. Fitzgerald, and N. Spruston:
"Learning produces a hippocampal cognitive map in the form of an orthogonalized state machine",
10.1101/2023.08.03.551900.

7 L. R. Squire:
Memory and the Hippocampus: A Synthesis From Findings With Rats, Monkeys, and Humans,
tech. rep. 2 (1992), pp. 195–231.

8 S. L. Brincat and E. K. Miller:
"Frequency-specific hippocampal-prefrontal interactions during associative learning",
Nature Neuroscience **18**, 576–581 (2015).

9 E. C. Tolman:
"Cognitive maps in rats and men.",
Psychological Review **55**, 189–208 (1948).

10 D. Farzanfar, H. J. Spiers, M. Moscovitch, and R. S. Rosenbaum:
From cognitive maps to spatial schemas,
2023.

11 C. Eden:
Cognitive mapping,
tech. rep. (1988), pp. 1–13.

12 D. Aronov, R. Nevers, and D. W. Tank:
"Mapping of a non-spatial dimension by the hippocampal-entorhinal circuit",
Nature **543**, 719–722 (2017).

13 P. Dimaggio:
CULTURE AND COGNITION,
tech. rep. (1997), pp. 263–87.

14 I. P. Kan, R. S. Rosenbaum, and M. Verfaellie:
Schema processing across the lifespan: From theory to applications,
2020.

15 P. P. Raykov, J. L. Keidel, J. Oakhill, and C. M. Bird:
"The brain regions supporting schema-related processing of people's identities",
Cognitive Neuropsychology **37**, 8–24 (2020).

16 N. J. Nersessian:
"In the Theoretician's Laboratory: Thought Experimenting as Mental Modeling",
PSA: Proceedings of the Biennial Meeting of the Philosophy of Science Association **1992**, 291–301 (1992).

17 F. P. de Lange, M. Heilbron, and P. Kok:
How Do Expectations Shape Perception?,
2018.

18 M. Kawato:
"Internal models for motor control and trajectory planning",
Current Opinion in Neurobiology **9**, 718–727 (1999).

19 G. Corrado and K. Doya:
Understanding neural coding through the model-based analysis of decision making,
2007.

20 K. J. Miller, M. Eckstein, M. M. Botvinick, and Z. Kurth-Nelson:
"Cognitive Model Discovery via Disentangled RNNs",
10.1101/2023.06.23.546250.

21 H. F. Harlow:
"The formation of learning sets.",
Psychological Review **56**, 51–65 (1949).

22 D. Tse, R. F. Langston, M. Kakeyama, I. Bethus, P. A. Spooner, E. R. Wood, M. P. Witter, and R. G. Morris:
"Schemas and memory consolidation",
Science **316**, 76–82 (2007).

23 J. Zhou, C. Jia, M. Montesinos-cartagena, and M. P. H. Gardner:
"Evolving schema representations in orbitofrontal ensembles during learning",
Nature, 10.1038/s41586-020-03061-2 (2020).

24 R. G. Morris:
Elements of a neurobiological theory of hippocampal function: The role of synaptic plasticity, synaptic tagging and schemas,
2006.

25 K. Kansky, T. Silver, D. A. Mély, M. Eldawy, M. Lázaro-Gredilla, X. Lou, N. Dorfman, S. Sidor, S. Phoenix, and D. George:
"Schema networks: Zero-shot transfer with a generative causal model of physics intuitive",
34th International Conference on Machine Learning, ICML 2017 **4**, 2879–2889 (2017).

26 V. Goudar, B. Peysakhovich, D. J. Freedman, E. A. Buffalo, and X.-J. Wang:
"Schema formation in a neural population subspace underlies learning-to-learn in flexible sensorimotor problem-solving",
Nature Neuroscience **26**, 879–890 (2023).

27 J. C. Whittington, D. McCaffary, J. J. Bakermans, and T. E. Behrens:
How to build a cognitive map,
2022.

28 E. T. Rolls:
"The functions of the orbitofrontal cortex",
Brain and Cognition **55**, 11–29 (2004).

29 H. Schultz, T. Sommer, and J. Peters:
The role of the human entorhinal cortex in a representational account of memory,
2015.

30 M. P. Witter, T. P. Doan, B. Jacobsen, E. S. Nilssen, and S. Ohara:
Architecture of the entorhinal cortex a review of entorhinal anatomy in rodents with some comparative notes,
2017.

31 Mark S, Baram A, Schwartenbeck P, Hahamy A, Samborska V, and Behrens T:
"Flexible and abstract neural representations of structural knowledge",
10.1101/2023.08.31.555760.

32 M. Sarafyazd and M. Jazayeri:
"Hierarchical reasoning by neural circuits in the frontal cortex",
Science **364**, 10.1126/science.aav8911 (2019).

33 G. Buzsáki:
"Theta Oscillations in the Hippocampus",
Neuron **33**, 325–340 (2002).

34 S. Bernardi, M. K. Benna, M. Rigotti, J. Munuera, S. Fusi, and C. D. Salzman:
"Updated - The geometry of abstraction in hippocampus and prefrontal cortex",
bioRxiv, 408633 (2018).

35 J. O'Keefe and J. Dostrovsky:
"The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat",
Brain Research **34**, 171–175 (1971).

36 D. Bush and N. Burgess:
"Advantages and detection of phase coding in the absence of rhythmicity",
Hippocampus, 1–18 (2020).

37 C. Geisler, D. Robbe, M. Zugaro, A. Sirota, and G. rgy Buzsá ki:
Hippocampal place cell assemblies are speed-controlled oscillators,
tech. rep. (2007).

38 J. R. Climer, E. L. Newman, and M. E. Hasselmo:
"Phase Coding By Grid Cells In Unconstrained Environments: Two-Dimensional Phase Precession",
European Journal of Neuroscience **38**, 2526–2541 (2013).

39 P. Fries:
"Rhythms for Cognition: Communication through Coherence",
Neuron **88**, 220–235 (2015).

40 H. T. Blair, A. Wu, and J. Cong:
"Oscillatory neurocomputing with ring attractors: A network architecture for mapping locations in space onto patterns of neural synchrony",
Philosophical Transactions of the Royal Society B: Biological Sciences **369**, 10.1098/rstb.2012.0526 (2014).

41 J. Orchard:
Oscillator-interference models of path integration do not require theta oscillations,
2015.

42 T. Eliav, M. Geva-Sagiv, M. M. Yartsev, A. Finkelstein, A. Rubin, L. Las, and N. Ulanovsky:
"Nonoscillatory Phase Coding and Synchronization in the Bat Hippocampal Formation",
Cell **175**, 1119–1130 (2018).

43 M. J. Jutras, P. Fries, and E. A. Buffalo:
"Oscillatory activity in the monkey hippocampus during visual exploration and memory formation",
Proceedings of the National Academy of Sciences of the United States of America **110**, 13144–13149 (2013).

44 D. S. Rizzuto, J. R. Madsen, E. B. Bromfield, A. Schulze-Bonhage, D. Seelig, R. Aschenbrenner-Scheibe, and M. J. Kahana:
"Reset of human neocortical oscillations during a working memory task",
Proceedings of the National Academy of Sciences **100**, 7931–7936 (2003).

45 C. M. Bird and N. Burgess:
The hippocampus and memory: Insights from spatial processing,
2008.

46 J. Lisman, G. Buzsáki, H. Eichenbaum, L. Nadel, C. Ranganath, and A. D. Redish:
Viewpoints: How the hippocampus contributes to memory, navigation and cognition,
2017.

47 R. D. Rubin, P. D. Watson, M. C. Duff, and N. J. Cohen:
The role of the hippocampus in flexible cognition and social behavior,
2014.

48 L. Nadel and M. Moscovitcht:
"Memory consolidation, retrograde amnesia and the hippocampal complex",
Current Opinion in Neurobiology **7**, 217–227 (1997).

49 N. P. Friedman and T. W. Robbins:
The role of prefrontal cortex in cognitive control and executive function,
2022.

50 S. M. Kolk and P. Rakic:
Development of prefrontal cortex,
2022.

51 J. Tanji and E. Hoshi:
"Role of the Lateral Prefrontal Cortex in Executive Behavioral Control",
10.1152/physrev.00014.2007.-The (2008).

52 D. R. Euston, A. J. Gruber, and B. L. McNaughton:
The Role of Medial Prefrontal Cortex in Memory and Decision Making,
2012.

53 R. Bartolo and B. B. Averbeck:
"Prefrontal Cortex Predicts State Switches during Reversal Learning",
Neuron **106**, 1044–1054 (2020).

54 F. A. Mansouri, K. Matsumoto, and K. Tanaka:
"Prefrontal cell activities related to monkeys' success and failure in adapting to rule changes in a Wisconsin card sorting test analog",
Journal of Neuroscience **26**, 2745–2756 (2006).

55 A. Rouzitalab, C. B. Boulay, J. Park, J. C. Martinez-Trujillo, and A. J. Sachs:
"Ensembles code for associative learning in the primate lateral prefrontal cortex",
Cell Reports **42**, 10.1016/j.celrep.2023.112449 (2023).

56 S. Funahashi, C. J. Bruce, and P. S. Goldman-Rakic:
Mnemonic Coding of Visual Space in the Monkey's Dorsolateral Prefrontal Cortex,
tech. rep. 2 (1989).

57 E. K. Miller, C. A. Erickson, and R. Desimone:
Neural Mechanisms of Visual Working Memory in Prefrontal Cortex of the Macaque,
tech. rep. (1996).

58 A. F. Rossi, L. Pessoa, R. Desimone, and L. G. Ungerleider:
"The prefrontal cortex and the executive control of attention",
in Experimental brain research, Vol. 192, 3 (2009), pp. 489–497.

59 M. F. Rushworth, M. A. P. Noonan, E. D. Boorman, M. E. Walton, and T. E. Behrens:
Frontal Cortex and Reward-Guided Learning and Decision-Making,
2011.

60 M. M. Botvinick, J. D. Cohen, and C. S. Carter:
Conflict monitoring and anterior cingulate cortex: An update,
2004.

61 D. Dignath, A. B. Eder, M. Steinhauser, and A. Kiesel:
Conflict monitoring and the affective-signaling hypothesis—An integrative review,
2020.

62 P. W. Frankland, B. Bontempi, L. E. Talton, L. Kaczmarek, and A. J. Silva:
"The Involvement of the Anterior Cingulate Cortex in Remote Contextual Fear Memory",
Science **304**, 881–883 (2004).

63 A. Takashima, K. M. Petersson, F. Rutters, I. Tendolkar, O. Jensen, M. J. Zwarts, B. L. Mcnaughton, and G. Fernández:
Declarative memory consolidation in humans: A prospective functional magnetic resonance imaging study,
tech. rep. 3 (2006), pp. 756–761.

64 C. W. Hoy, D. R. Quiroga-Martinez, D. King-Stephens, K. D. Laxer, P. Weber, J. J. Lin, and R. T. Knight:
"Asymmetric coding of reward prediction errors in human insula and dorsomedial prefrontal cortex",
NeuroImage **263**, 10.1101/2022.12.07.519496 (2022).

65 C. K. Starkweather, S. J. Gershman, and N. Uchida:
"The Medial Prefrontal Cortex Shapes Dopamine Reward Prediction Errors under State Uncertainty",
Neuron **98**, 616–629 (2018).

66 P. Domenech, S. Rheims, and E. Koechlin:
"Neural mechanisms resolving exploitation-exploration dilemmas in the medial prefrontal cortex",
Science **369**, 10.1126/science.abb0184 (2020).

67 M. L. Kringelbach:
"The human orbitofrontal cortex: linking reward to hedonic experience",
Nature Reviews Neuroscience **6**, 691–702 (2005).

68 J. O Doherty, E. T. Rolls, S. Francis, R. Bowtell, F. McGlone, G. Kobal, B. Renner, and G. Ahne:
"Sensory-specific satiety-related olfactory activation of the human orbitofrontal cortex",
NeuroReport **11**, 399–403 (2000).

69 J. A. Gottfried, J. O'Doherty, and R. J. Dolan:
"Encoding Predictive Reward Value in Human Amygdala and Orbitofrontal Cortex",
Science **301**, 1104–1107 (2003).

70 D. M. Small, D. H. Zald, M. Jones-Gotman, R. J. Zatorre, J. V. Pardo, S. Frey, and M. Petrides:
"Human cortical gustatory areas",
NeuroReport **10**, 7–13 (1999).

71 E. Rolls:
"Representations of Pleasant and Painful Touch in the Human Orbitofrontal and Cingulate Cortices",
Cerebral Cortex **13**, 308–317 (2003).

72 G. Thut, W. Schultz, U. Roelcke, M. Nienhusmeier, J. Missimer, R. P. Maguire, and K. L. Leenders:
"Activation of the human brain by monetary reward",
NeuroReport **8**, 1225–1228 (1997).

73 S. H. Wang and R. G. Morris:
"Hippocampal-neocortical interactions in memory formation, consolidation, and reconsolidation",
Annual Review of Psychology **61**, 49–79 (2010).

74 L. R. Squire, L. Genzel, J. T. Wixted, and R. G. Morris:
"Memory consolidation",
Cold Spring Harbor Perspectives in Biology **7**, 10.1101/cshperspect.a021766 (2015).

75 H. Eichenbaum:
Prefrontal-hippocampal interactions in episodic memory,
2017.

76 D. L. Rosene and G. W. Van Hoesen:
"Hippocampal efferents reach widespread areas of cerebral cortex and amygdala in the rhesus monkey",
Science **198**, 315–317 (1977).

77 P. Lavenex, W. A. Suzuki, and D. G. Amaral:
"Perirhinal and parahippocampal cortices of the macaque monkey: Projections to the neocortex",
Journal of Comparative Neurology **447**, 394–420 (2002).

78 A. G. Siapas, E. V. Lubenov, and M. A. Wilson:
"Prefrontal phase locking to hippocampal theta oscillations",
Neuron **46**, 141–151 (2005).

79 M. L. Schlichting and A. R. Preston:
"Hippocampal–medial prefrontal circuit supports memory updating during learning and post-encoding rest",
Neurobiology of Learning and Memory **134**, 91–106 (2016).

80 J. Minxha, R. Adolphs, S. Fusi, A. N. Mamelak, and U. Rutishauser:
"Flexible recruitment of memory-based choice representations by the human medial frontal cortex",
Science **368**, 10.1126/science.aba3313 (2020).

81 A. Das and V. Menon:
"Asymmetric frequency-specific feedforward and feedback information flow between hippocampus and prefrontal cortex during verbal memory encoding and recall",
Journal of Neuroscience **41**, 8427–8440 (2021).

82 S. Fusi, W. F. Asaad, E. K. Miller, and X. J. Wang:
"A Neural Circuit Model of Flexible Sensorimotor Mapping: Learning and Forgetting on Multiple Timescales",
Neuron **54**, 319–333 (2007).

83 F. Bouchacourt, S. Tafazoli, M. Mattar, T. J. Buschman, and N. D. Daw:
"Fast rule switching and slow rule updating in a perceptual categorization task",
eLife **11**, 10.7554/elife.82531 (2022).

84 N. Cowan:
"The magical mystery four: How is working memory capacity limited, and why?",
Current Directions in Psychological Science **19**, 51–57 (2010).

85 N. Cowan:
"The magical number 4 in short-term memory: A reconsideration of mental storage capacity",
Behavioral and Brain Sciences **24**, 87–114 (2001).

86 P. Yurt, A. Calapai, R. Mundry, and S. Treue:
"Assessing cognitive flexibility in humans and rhesus macaques with visual motion and neutral distractors",
Frontiers in Psychology **13**, 10.3389/fpsyg.2022.1047292 (2022).

87 L. C. Elmore and A. A. Wright:
"Monkey visual short-term memory directly compared to humans",
Journal of Experimental Psychology: Animal Learning and Cognition **41**, 32–38 (2015).

88 J. J. Chelonis, A. R. Cox, M. J. Karr, P. K. Prunty, R. L. Baldwin, and M. G. Paule:
"Comparison of delayed matching-to-sample performance in monkeys and children",
Behavioural Processes **103**, 261–268 (2014).

89 F. Barcelo, M. Sanz, V. Molina, and F. J. Rubia:
The Wisconsin Card Sorting Test and the assessment of frontal function] A validation study with event!related potentials,
tech. rep. (1997).

90 S. Konishi:
"Contribution of Working Memory to Transient Activation in Human Inferior Prefrontal Cortex during Performance of the Wisconsin Card Sorting Test",
Cerebral Cortex **9**, 745–753 (1999).

91 M. Nowak and K. Slgmundt:
7. Colson, R.0,
tech. rep. 5 (1978), pp. 135–150.

92 J. Hogeveen, T. S. Mullins, J. D. Romero, E. Eversole, K. Rogge-Obando, A. R. Mayer, and V. D. Costa:
"The neurocomputational bases of explore-exploit decision-making",
Neuron **110**, 1869–1879 (2022).

93 A. C. Costa, T. Ahamed, and G. J. Stephens:
"Adaptive, locally linear models of complex dynamics",
Proceedings of the National Academy of Sciences of the United States of America **116**, 1501–1510 (2019).

94 Y. Niv, R. Daniel, A. Geana, S. J. Gershman, Y. C. Leong, A. Radulescu, and R. C. Wilson:
"Reinforcement learning in multidimensional environments relies on attention mechanisms",
Journal of Neuroscience **35**, 8145–8157 (2015).

95 D. Grant and E. Berg:
"Wisconsin Card Sorting Test",
Journal of Experimental Psychology, 10.1037/t31298-000 (1948).

96 A. A. Wright:
"AN EXPERIMENTAL ANALYSIS OF MEMORY PROCESSING",
Journal of the Experimental Analysis of Behavior **88**, 405–433 (2007).

97 A. L. Robinson, R. K. Heaton, R. A. Lehman, and D. W. Stilson:
"The utility of the Wisconsin Card Sorting Test in detecting and localizing frontal lobe lesions.",
Journal of Consulting and Clinical Psychology **48**, 605–614 (1980).

98 S. D. Muthukumaraswamy:
High-frequency brain activity and muscle artifacts in MEG/EEG: A review and recommendations,
2013.

99 A. L. Hodgkin and A. F. Huxley:
"A quantitative description of membrane current and its application to conduction and excitation in nerve",
The Journal of Physiology **117**, 500–544 (1952).

100 G. Buzsáki, C. A. Anastassiou, and C. Koch:
"The origin of extracellular fields and currents-EEG, ECoG, LFP and spikes",
Nature Reviews Neuroscience **13**, 407–420 (2012).

101 R. Elul:
"The Genesis of the Eeg",
in (1972), pp. 227–272.

102 G. T. Einevoll, C. Kayser, N. K. Logothetis, and S. Panzeri:
"Modelling and analysis of local field potentials for studying the function of cortical circuits",
Nature Reviews Neuroscience **14**, 770–785 (2013).

103 R. LORENTE de NO:
"A study of nerve physiology.",
Studies from the Rockefeller institute for medical research. Reprints. Rockefeller Institute for Medical Research **131**, 1–496 (1947).

104 Z. Shi, D. M. Wilkes, P. F. Yang, F. Wang, R. Wu, T. L. Wu, L. M. Chen, and J. C. Gore:
"On the Relationship between MRI and Local Field Potential Measurements of Spatial and Temporal Variations in Functional Connectivity",
Scientific Reports **9**, 10.1038/s41598-019-45404-8 (2019).

105 M. Steriade, A. Nuiiez, and F. Amzica:
Intracellular Analysis of Relations between the Slow (<I Hz) Neocortical Oscillation and Other Sleep Rhythms of the Electroencephalogram,
tech. rep. 8 (1993), pp. 3288–3283.

106 J. Lisman:
"The theta/gamma discrete phase code occuring during the hippocampal phase precession may be a more general brain coding scheme",
Hippocampus **15**, 913–922 (2005).

107 P. Sauseng, W. Klimesch, M. Doppelmayr, S. Hanslmayr, M. Schabus, and W. R. Gruber:
"Theta coupling in the human electroencephalogram during a working memory task",
Neuroscience Letters **354**, 123–126 (2004).

108 O. Jensen and A. Mazaheri:
"Shaping functional architecture by oscillatory alpha activity: Gating by inhibition",
Frontiers in Human Neuroscience **4**, 10.3389/fnhum.2010.00186 (2010).

109 W. Klimesch:
Alpha-band oscillations, attention, and controlled access to stored information,
2012.

110 A. Eusebio and P. Brown:
"Oscillatory activity in the basal ganglia",
Parkinsonism and Related Disorders **13**, 10.1016/S1353-8020(08)70044-0 (2007).

111 R. Schmidt, M. H. Ruiz, B. E. Kilavik, M. Lundqvist, P. A. Starr, and A. R. Aron:
"Beta oscillations in working memory, executive control of movement and thought, and sensorimotor function",
in Journal of neuroscience, Vol. 39, 42 (2019), pp. 8231–8238.

112 B. E. Kilavik, M. Zaepffel, A. Brovelli, W. A. MacKay, and A. Riehle:
The ups and downs of beta oscillations in sensorimotor cortex,
2013.

113 C. Tallon-Baudry, O. Bertrand, C. Delpuech, and J. Pernier:
Oscillatory-Band (30-70 Hz) Activity Induced by a Visual Search Task in Humans,
tech. rep. (1997).

114 G. Buzśaki and X. J. Wang:
Mechanisms of gamma oscillations,
2012.

115 S. Ray, N. E. Crone, E. Niebur, P. J. Franaszczuk, and S. S. Hsiao:
"Neural correlates of high-gamma oscillations (60-200 Hz) in macaque local field potentials and their potential implications in electrocorticography",
Journal of Neuroscience **28**, 11526–11536 (2008).

116 K. J. Miller, L. B. Sorensen, J. G. Ojemann, and M. Den Nijs:
"Power-law scaling in the brain surface electric potential",
PLoS Computational Biology **5**, 10.1371/journal.pcbi.1000609 (2009).

117 K. H. Pettersen and G. T. Einevoll:
"Amplitude variability and extracellular low-pass filtering of neuronal spikes",
Biophysical Journal **94**, 784–802 (2008).

118 H. G. Rey, C. Pedreira, and R. Quian Quiroga:
Past, present and future of spike sorting techniques,
2015.

119 S. Waldert, R. N. Lemon, and A. Kraskov:
"Influence of spiking activity on cortical local field potentials",
Journal of Physiology **591**, 5291–5303 (2013).

120 C. A. Bosman, J. M. Schoffelen, N. Brunet, R. Oostenveld, A. M. Bastos, T. Womelsdorf, B. Rubehn, T. Stieglitz, P. De Weerd, and P. Fries:
"Attentional Stimulus Selection through Selective Synchronization between Monkey Visual Areas",
Neuron **75**, 875–888 (2012).

121 A. Dubey and S. Ray:
"Spatial spread of local field potential is band-pass in the primary visual cortex",
J Neurophysiol **116**, 10.1152/jn.00443.2016.-Local (2016).

122 G. H. Glover:
Overview of functional magnetic resonance imaging,
2011.

123 C. E. Collins, D. C. Airey, N. A. Young, D. B. Leitch, and J. H. Kaas:
"Neuron densities vary across and within cortical areas in primates",
Proceedings of the National Academy of Sciences of the United States of America **107**, 15927–15932 (2010).

124 M. S. Franklin, G. W. Kraemer, S. E. Shelton, E. Baker, N. H. Kalin, and H. Uno:
Gender differences in brain volume and size of corpus callosum and amygdala of rhesus monkey measured from MRI images,
tech. rep. (2000), pp. 263–267.

125 J. P. Lachaux, D. Rudrauf, and P. Kahane:
"Intracranial EEG and human brain mapping",
in Journal of physiology paris, Vol. 97, 4-6 (2003), pp. 613–628.

126 P. Ablin, J. F. Cardoso, and A. Gramfort:
"Faster independent component analysis by preconditioning with hessian approximations",
IEEE Transactions on Signal Processing **66**, 4040–4049 (2018).

127 F. Campos Viola, J. Thorne, B. Edmonds, T. Schneider, T. Eichele, and S. Debener:
"Semi-automatic identification of independent components representing EEG artifact",
Clinical Neurophysiology **120**, 868–877 (2009).

128 X. Jiang, G. B. Bian, and Z. Tian:
Removal of artifacts from EEG signals: A review,
2019.

129 D. J. Thomson:
"Jackknifing multitaper spectrum estimates",
IEEE Signal Processing Magazine **24**, 20–30 (2007).

130 S. E. Kim, M. K. Behr, D. Ba, and E. N. Brown:
"State-space multitaper time-frequency analysis",
Proceedings of the National Academy of Sciences of the United States of America **115**, E5–E14 (2018).

131 N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. Chao Tung, and H. H. Liu:
The Empirical Mode Decomposition and the Hilbert Spectrum for Nonlinear and Non-Stationary Time Series Analysis,
tech. rep. (1998), pp. 903–995.

132 Z. Wu and N. E. Huang:
Ensemble empirical mode decomposition: A noise-assisted data analysis method,
2009.

133 V. V. Moca, H. Bârzan, A. Nagy-Dăbâcan, and R. C. Mureșan:
"Time-frequency super-resolution with superlets",
Nature Communications **12**, 10.1038/s41467-020-20539-9 (2021).

134 Y.-W. Liu:
"Hilbert Transform and Applications",
in *Fourier transform applications* (InTech, 2012).

135 M. X. Cohen:
"A better way to define and describe Morlet wavelets for time-frequency analysis",
NeuroImage **199**, 81–86 (2019).

136 Kluyver and JC:
Huygens Institute-Royal Netherlands Academy of Arts and Sciences (KNAW),
tech. rep. (1905).

137 J. A. Greenwood and D. Durand:
The Distribution of Length and Components of the Sum of n Random Unit Vectors,
tech. rep. 2 (1955), pp. 233–246.

138 K. PEARSON:
"The Problem of the Random Walk",
Nature **72**, 294–294 (1905).

139 T. Sakane, S. Okabayashi, S. Kimura, D. Inoue, A. Tanaka, and T. Furubayashi:
"Brain and nasal cavity anatomy of the cynomolgus monkey: Species differences from the viewpoint of direct delivery from the nose to the brain",
Pharmaceutics **12**, 1–8 (2020).

140 S. Thorpe and M. Fabre-Thorpe:
"Seeking_Categories_in_the_Brai",
Science **291**, 260–263 (2001).

141 K. D. Harris:
"Nonsense correlations in neuroscience",
10.1101/2020.11.29.402719.

142 H. M. Vergara, K. Harris, A. Kulkarni, and G. T. Meijer:
"Open Peer-Review Open Data Open Code Neurons in the mouse brain correlate with cryptocurrency price: a cautionary tale",
10.31234/osf.io/fa4wz.

143 M. Vinck, R. Batista-Brito, U. Knoblich, and J. A. Cardin:
"Arousal and Locomotion Make Distinct Contributions to Cortical Activity Patterns and Visual Encoding",
Neuron **86**, 740–754 (2015).

144 A. V. Goonawardena, S. R. Morairty, G. A. Orellana, A. R. Willoughby, T. L. Wallace, and T. S. Kilduff:
"Electrophysiological characterization of sleep/wake, activity and the response to caffeine in adult cynomolgus macaques",
Neurobiology of Sleep and Circadian Rhythms **6**, 9–23 (2019).

145 D. Kobak, W. Brendel, C. Constantinidis, C. E. Feierstein, A. Kepecs, Z. F. Mainen, X. L. Qi, R. Romo, N. Uchida, and C. K. Machens:
"Demixed principal component analysis of neural population data",
eLife **5**, 1–36 (2016).

146 S. Ray and J. H. Maunsell:
"Different origins of gamma rhythm and high-gamma activity in macaque visual cortex",
PLoS Biology **9**, 10.1371/journal.pbio.1000610 (2011).

147 V. Goudar, J.-W. Kim, Y. Liu, A. J. O Dede, M. J. Jutras, I. Skelin, M. Ruvalcaba, W. Chang, A. L. Fairhall, J. J. Lin, R. T. Knight, and X.-J. Wang:
"Comparing rapid rule-learning strategies in humans",
10.1101/2023.01.10.523416.

148 N. D. Daw, J. P. O'Doherty, P. Dayan, B. Seymour, and R. J. Dolan:
"Cortical substrates for exploratory decisions in humans",
Nature **441**, 876–879 (2006).

149 M. Ito and K. Doya:
"Validation of decision-making models and analysis of decision variables in the rat basal ganglia",
Journal of Neuroscience **29**, 9861–9874 (2009).

150 A. Bari, D. E. Theobald, D. Caprioli, A. C. Mar, A. Aidoo-Micah, J. W. Dalley, and T. W. Robbins:
"Serotonin modulates sensitivity to reward and negative feedback in a probabilistic reversal learning task in rats",
Neuropsychopharmacology **35**, 1290–1301 (2010).

151 G. L. Dalton, A. G. Phillips, and S. B. Floresco:
"Preferential involvement by nucleus accumbens shell in mediating probabilistic learning and reversal shifts",
Journal of Neuroscience **34**, 4618–4626 (2014).

152 K. Swanson, B. B. Averbeck, and M. Laubach:
"Noradrenergic regulation of Win-Stay/Lose-Shift policy and choice determinism in a two-armed bandit task",
10.1101/2020.11.13.382069.

153 N. Patel, L. Acerbi, and A. Pouget:
Dynamic allocation of limited memory resources in reinforcement learning,
tech. rep. ().

154 Q. Lan, Y. Pan, J. Luo, and A. R. Mahmood:
"Memory-efficient Reinforcement Learning with Value-based Knowledge Consolidation",
(2022).

155 A. J. Bishara, J. K. Kruschke, J. C. Stout, A. Bechara, D. P. McCabe, and J. R. Busemeyer:
"Sequential learning models for the Wisconsin card sort task: Assessing processes in substance dependent individuals",
Journal of Mathematical Psychology **54**, 5–13 (2010).

156 L. Rabiner:
"A tutorial on hidden Markov models and selected applications in speech recognition",
Proceedings of the IEEE **77**, 257–286 (1989).

157 F. Gao and L. Han:
"Implementing the Nelder-Mead simplex algorithm with adaptive parameters",
Computational Optimization and Applications **51**, 259–277 (2012).

158 V. D. Costa, V. L. Tran, J. Turchi, and B. B. Averbeck:
"Reversal learning and dopamine: A Bayesian perspective",
Journal of Neuroscience **35**, 2407–2416 (2015).

159 H. M. Bayer and P. W. Glimcher:
"Midbrain Dopamine Neurons Encode a Quantitative Reward Prediction Error Signal",
Neuron **47**, 129–141 (2005).

160 N. D. Daw and P. N. Tobler:
"Value Learning through Reinforcement: The Basics of Dopamine and Reinforcement Learning",
in *Neuroeconomics: decision making and the brain: second edition* (Elsevier Inc., 2013), pp. 283–298.

161 J. W. Pillow, J. Shlens, L. Paninski, A. Sher, A. M. Litke, E. J. Chichilnisky, and E. P. Simoncelli:
"Spatio-temporal correlations and visual signalling in a complete neuronal population",
Nature **454**, 995–999 (2008).

162 D. J. Ottenheimer, M. M. Hjort, A. J. Bowen, N. A. Steinmetz, and G. D. Stuber:
A stable, distributed code for cue value in mouse cortex during reward learning,
tech. rep. (2023), p. 84604.

163 A. C. Van Duijvenvoorde, K. Zanolie, S. A. Rombouts, M. E. Raijmakers, and E. A. Crone:
"Evaluating the negative or valuing the positive? Neural mechanisms supporting feedback-based learning across development",
Journal of Neuroscience **28**, 9495–9503 (2008).

164 S. Peters, A. C. Van Duijvenvoorde, P. C. M. Koolschijn, and E. A. Crone:
"Longitudinal development of frontoparietal activity during feedback learning: Contributions of age, performance, working memory and cortical thickness",
Developmental Cognitive Neuroscience **19**, 211–222 (2016).

165 I. Van De Vijver, K. R. Ridderinkhof, and M. X. Cohen:
Frontal Oscillatory Dynamics Predict Feedback Learning and Action Adjustment,
tech. rep. (2011).

166 M. Rigotti, O. Barak, M. R. Warden, X. J. Wang, N. D. Daw, E. K. Miller, and S. Fusi:
"The importance of mixed selectivity in complex cognitive tasks",
Nature **497**, 585–590 (2013).

167 F. Mormann, J. Fell, N. Axmacher, B. Weber, K. Lehnertz, C. E. Elger, and G. Fernández:
"Phase/amplitude reset and theta-gamma interaction in the human medial temporal lobe during a continuous word recognition memory task",
Hippocampus **15**, 890–900 (2005).

168 B. Givens:
"Stimulus-evoked resetting of the dentate theta rhythm",
NeuroReport **8**, 159–163 (1996).

169 J. M. Williams and B. Givens:
Stimulation-induced reset of hippocampal theta in the freely performing rat,
2003.

170 P. T. Huerta and J. E. Lisman:
"Heightened synaptic plasticity of hippocampal CA1 neurons during a Cholinergically induced rhythmic state",
Nature **364**, 723–725 (1993).

171 P. T. Huerta and J. E. Lisman:
Bidirectional Synaptic Plasticity Induced by a Single Burst during Cholinergic Theta Oscillation in CA1 In Vitro,
tech. rep. (1995), pp. 1053–1063.

172 C. Hö Lscher, R. Anwyl, and M. J. Rowan:
Stimulation on the Positive Phase of Hippocampal Theta Rhythm Induces Long-Term Potentiation That Can Be Depotentiated by Stimulation on the Negative Phase in Area CA1 In Vivo,
tech. rep. (1997).

173 W. Klimesch, P. Sauseng, S. Hanslmayr, W. Gruber, and R. Freunberger:
Event-related phase reorganization may explain evoked neural dynamics,
2007.

174 A. S. Shah, S. L. Bressler, K. H. Knuth, M. Ding, A. D. Mehta, I. Ulbert, and C. E. Schroeder:
"Neural Dynamics and the Fundamental Mechanisms of Event-related Brain Potentials",
Cerebral Cortex **14**, 476–483 (2004).

175 P. Sauseng, W. Klimesch, W. R. Gruber, S. Hanslmayr, R. Freunberger, and M. Doppelmayr:
Are event-related potential components generated by phase resetting of brain oscillations? A critical discussion,
2007.

176 I. E. Monosov, J. C. Trageser, and K. G. Thompson:
"Measurements of Simultaneously Recorded Spiking Activity and Local Field Potentials Suggest that Spatial Selection Emerges in the Frontal Eye Field",
Neuron **57**, 614–625 (2008).

177 M. Angjelichinoski, J. Choi, T. Banerjee, B. Pesaran, and V. Tarokh:
"Cross-subject decoding of eye movement goals from local field potentials",
in Journal of neural engineering, Vol. 17, 1 (2020).

178 D. Sussillo, R. Jozefowicz, L. F. Abbott, and C. Pandarinath:
"LFADS - Latent Factor Analysis via Dynamical Systems",
(2016).

179 Z. Ghahramani and G. E. Hinton:
Communicated by Volker Tresp Variational Learning for Switching State-Space Models,
tech. rep. (2000).

180 E. B. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky:
Nonparametric Bayesian Learning of Switching Linear Dynamical Systems,
tech. rep. (2008).

181 J. I. Glaser, M. Whiteway, J. P. Cunningham, L. Paninski, and S. W. Linderman:
"Recurrent switching dynamical systems models for multiple interacting neural populations",
bioRxiv, 2020.10.21.349282 (2020).

182 C. Pandarinath, D. J. O'Shea, J. Collins, R. Jozefowicz, S. D. Stavisky, J. C. Kao, E. M. Trautmann, M. T. Kaufman, S. I. Ryu, L. R. Hochberg, J. M. Henderson, K. V. Shenoy, L. F. Abbott, and D. Sussillo:
"Inferring single-trial neural population dynamics using sequential auto-encoders",
Nature Methods **15**, 805–815 (2018).

183 R. Srinath and S. Ray:
"First published April 30",
J Neurophysiol **112**, 741–751 (2014).

184 A. Shojaie and E. B. Fox:
"Granger Causality: A Review and Recent Advances",
(2021).

185 A. K. Seth, A. B. Barrett, and L. Barnett:
"Granger causality analysis in neuroscience and neuroimaging",
Journal of Neuroscience **35**, 3293–3297 (2015).

186 W. J. Freeman, B. C. Burke, and M. D. Holmes:
"Aperiodic phase re-setting in scalp EEG of beta-gamma oscillations by state transitions at alpha-theta rates",
Human Brain Mapping **19**, 248–272 (2003).

187 T. Askham and J. N. Kutz:
"Variable projection methods for an optimized dynamic mode decomposition",
SIAM Journal on Applied Dynamical Systems **17**, 380–416 (2018).

188 D. Sashidhar and J. N. Kutz:
"Bagging, optimized dynamic mode decomposition for robust, stable forecasting with spatial and temporal uncertainty quantification",
Philosophical Transactions of the Royal Society A **380** (2022).

189 S. Maćešić, N. Črnjarić-Žic, and I. Mezić:
"Koopman operator family spectrum for nonautonomous systems",
SIAM Journal on Applied Dynamical Systems **17**, 2478–2515 (2018).

190 J. N. Kutz, X. Fu, and S. L. Brunton:
"Multiresolution dynamic mode decomposition",
SIAM Journal on Applied Dynamical Systems **15**, 713–735 (2016).

191 B. R. Noack, K. Afanasiev, M. Morzyński, G. Tadmor, and F. Thiele:
"A hierarchy of low-dimensional models for the transient and post-transient cylinder wake",
Journal of Fluid Mechanics **497**, 335–363 (2003).

192 A. Provenzale:
"Climate as a Complex Dynamical System",
in (2014), pp. 135–142.

193 S. Ananthakrishnan, P. Geetha, and K. P. Soman:
"Temperature forecasting using dynamic mode decomposition",
in Proceedings of the 3rd international conference on intelligent sustainable systems, iciss 2020 (2020), pp. 1590–1596.

194 H. S. Seung:
How the brain keeps the eyes still,
tech. rep. (1996), pp. 13339–13344.

195 K. Zhang:
Representation of Spatial Orientation by the Intrinsic Dynamics of the Head-Direction Cell Ensemble: A Theory,
tech. rep. 6 (1996), pp. 2112–2126.

196 J. J. Hopfield:
"Neural networks and physical systems with emergent collective computational abilities.",
Proceedings of the National Academy of Sciences of the United States of America **79**, 2554–2558 (1982).

197 K. D. Harris, A. Aravkin, R. Rao, and B. W. Brunton:
"Time-Varying Autoregression with Low-Rank Tensors",
SIAM Journal on Applied Dynamical Systems **20**, 2335–2358 (2021).

198 P. J. Schmid:
"Dynamic mode decomposition of numerical and experimental data",
Journal of Fluid Mechanics **656**, 5–28 (2010).

199 C. W. Rowley, I. Mezi, S. Bagheri, P. Schlatter, and D. S. Henningson:
"Spectral analysis of nonlinear flows",
Journal of Fluid Mechanics **641**, 115–127 (2009).

200 B. O. Koopman and J. v. Neumann:
"Dynamical Systems of Continuous Spectra",
Proceedings of the National Academy of Sciences **18**, 255–263 (1932).

201 S. L. Brunton, M. Budišić, E. Kaiser, and J. N. Kutz:
"Modern Koopman Theory for Dynamical Systems",
(2021).

202 G. Snyder and Z. Song:
"Koopman Operator Theory for Nonlinear Dynamic Modeling using Dynamic Mode Decomposition",
(2021).

203 T. Gallagher, T. Bjorness, R. Greene, Y. J. You, and L. Avery:
"The Geometry of Locomotive Behavioral States in C. elegans",
PLoS ONE **8**, 10.1371/journal.pone.0059865 (2013).

204 A. H. Song, D. Ba, and E. N. Brown:
PLSO: A Generative Framework for Decomposing Nonstationary Time-Series into Piecewise Stationary Oscillatory Components,
tech. rep. ().

205 S. T. Dawson, M. S. Hemati, M. O. Williams, and C. W. Rowley:
"Characterizing and correcting for the effect of sensor noise in the dynamic mode decomposition",
Experiments in Fluids **57**, 10.1007/s00348-016-2127-7 (2016).

206 J. Page and R. R. Kerswell:
"Koopman mode expansions between simple invariant solutions",
Journal of Fluid Mechanics **879**, 1–27 (2019).

207 J. D. Bryan and S. E. Levinson:
"Autoregressive Hidden Markov Model and the Speech Signal",
in Procedia computer science, Vol. 61 (2015), pp. 328–333.

208 I. Stanculescu, C. K. Williams, and Y. Freer:
"Autoregressive Hidden Markov Models for the Early Detection of Neonatal Sepsis",
IEEE Journal of Biomedical and Health Informatics **18**, 1560–1570 (2014).

209 J. M. Haslbeck, L. F. Bringmann, and L. J. Waldorp:
"A Tutorial on Estimating Time-Varying Vector Autoregressive Models",
Multivariate Behavioral Research **56**, 120–149 (2020).

210 T. G. Kolda and B. W. Bader:
Tensor decompositions and applications,
2009.

211 J. H. Tu, C. W. Rowley, D. M. Luchtenburg, S. L. Brunton, and J. N. Kutz:
"On dynamic mode decomposition: Theory and applications",
Journal of Computational Dynamics **1**, 391–421 (2014).

212 M. O. Williams, M. S. Hemati, S. T. Dawson, I. G. Kevrekidis, and C. W. Rowley:
"Extending Data-Driven Koopman Analysis to Actuated Systems",
in Ifac-papersonline, Vol. 49, 18 (2016), pp. 704–709.

213 J. L. Proctor, S. L. Brunton, and J. N. Kutz:
"Dynamic mode decomposition with control",
SIAM Journal on Applied Dynamical Systems **15**, 142–161 (2016).

214 G. H. Golub and V. Pereyra:
The Differentiation of Pseudo-Inverses and Nonlinear Least Squares Problems Whose Variables Separate,
tech. rep. 2 (1973), pp. 413–432.

215 B. W. Brunton, L. A. Johnson, J. G. Ojemann, and J. N. Kutz:
"Extracting spatial-temporal coherent patterns in large-scale neural recordings using dynamic mode decomposition",
10.1016/j.jneumeth.2015.10.010 (2014).

216 N. Qian:
On the momentum term in gradient descent learning algorithms,
tech. rep. ().

217 V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos:
A review of feature selection methods on synthetic data,
2013.

218 P. Pudil and J. Novovi:
Pattern Recognition Letters Floating search methods in feature selection,
tech. rep. (1994).

219 B. J. He:
Scale-free brain activity: Past, present, and future,
2014.

220 B. E. Kilavik, A. Ponce-Alvarez, R. Trachel, J. Confais, S. Takerkart, and A. Riehle:
"Context-related frequency modulations of macaque motor cortical LFP beta oscillations",
Cerebral Cortex **22**, 2148–2159 (2012).

221 X. Xu, C. Zheng, and T. Zhang:
"Reduction in LFP cross-frequency coupling between theta and gamma rhythms associated with impaired STP and LTP in a rat model of brain ischemia",
Frontiers in Computational Neuroscience, 10.3389/fncom.2013.00027 (2013).

222 M. Esghaei, M. R. Daliri, and S. Treue:
"Attention decreases phase-amplitude coupling, enhancing stimulus discriminability in cortical area MT",
Frontiers in Neural Circuits **9**, 10.3389/fncir.2015.00082 (2015).

223 T. T. Munia and S. Aviyente:
"Time-Frequency Based Phase-Amplitude Coupling Measure For Neuronal Oscillations",
Scientific Reports **9**, 10.1038/s41598-019-48870-2 (2019).

224 N. Marrouch, J. Slawinska, D. Giannakis, and H. L. Read:
"Data-driven Koopman operator approach for computational neuroscience",
Annals of Mathematics and Artificial Intelligence **88**, 1155–1173 (2020).

225 J. Macqueen:
SOME METHODS FOR CLASSIFICATION AND ANALYSIS OF MULTIVARIATE OBSERVATIONS,
tech. rep. ().

226 P. D. Welch:
The Use of Fast Fourier Transform for the Estimation of Power Spectra: A Method Based on Time Aver. aging Over Short, Modified Periodograms Ik(fn) =-I A h (%) [a k,
tech. rep. 2 (1967).

227 R. W. Reynolds, N. A. Rayner, T. M. Smith, and D. C. Stokes:
An Improved In Situ and Satellite SST Analysis for Climate,
tech. rep. (2002).

228 M. Nolan, B. Pesaran, E. Shlizerman, and A. Orsborn:
"Multi-block RNN Autoencoders Enable Broadband ECoG Signal Reconstruction",
10.1101/2022.09.07.507004.

229 C.-J. Lin:
"Projected Gradient Methods for Non-negative Matrix Factorization",
Neural Computation, https://doi.org/10.1162/neco.2007.19.10.2756 (2007).

230 N. Pilnenskiy and I. Smetannikov:
"Feature selection algorithms as one of the python data analytical tools",
Future Internet **12**, 10.3390/fi12030054 (2020).

231 L. Ward and P. Greenwood:
"1/f noise",
Scholarpedia **2**, 1537 (2007).

232 G. G. Gregoriou, S. J. Gotts, H. Zhou, and R. Desimone:
"High-Frequency, long-range coupling between prefrontal and visual cortex during attention",
Science **324**, 1207–1210 (2009).